Spring 2013

TDMA Slot Reservation in Cluster-Based VANETs

Mohammad Salem Almalag
Old Dominion University

Recommended Citation

Almalag, Mohammad S.. "TDMA Slot Reservation in Cluster-Based VANETs" (2013). Doctor of Philosophy (PhD), dissertation, Computer Science, Old Dominion University, DOI: 10.25777/zjjg-ep34
https://digitalcommons.odu.edu/computerscience_etds/9

TDMA SLOT RESERVATION IN CLUSTER-BASED VANETS

by

Mohammad Salem Almalag
B.S. December 2000, King Saud University
M.S. December 2004, Ball State University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
May 2013

Approved by:

Dr. Michele C. Weigle (Director)

Dr. Kurt Maly (Member)

Dr. Hussein Abdel-Wahab (Member)

Dr. Stephan Olariu (Member)

Dr. Dimitrie C. Popescu (Member)

ABSTRACT

TDMA SLOT RESERVATION IN CLUSTER-BASED VANETS

Mohammad Salem Almalag
Old Dominion University, 2013
Director: Dr. Michele C. Weigle

Vehicular Ad Hoc Networks (VANETs) are a form of Mobile Ad Hoc Networks (MANETs) in which vehicles on the road form the nodes of the network. VANETs provide several services to enhance the safety and comfort of drivers and passengers. These services can be obtained by the wireless exchange of information among the vehicles driving on the road. In particular, the transmission of two different types of messages, safety/update and non-safety messages.

The transmission of safety/update message aims to inform the nearby vehicles about the sender's current status and/or a detected dangerous situation. This type of transmission is designed to help in accident and danger avoidance. Moreover, it requires high message generated rate and high reliability. On the other hand, the transmission of non-safety message aims to increase the comfort on vehicles by supporting several non-safety services, from notifications of traffic conditions to file sharing. Unfortunately, the transmission of non-safety message has less priority than safety messages, which may cause shutting down the comfort services. The goal of this dissertation is to design a MAC protocol in order to provide the ability of the transmission of non-safety message with little impact on the reliability of transmitting safety message even if the traffic and communication densities are high.

VANET is a highly dynamic network. With lack of specialized hardware for infrastructure and the mobility to support network stability and channel utilization, a cluster-based MAC protocol is needed to solve these overcomes.

This dissertation makes the following contributions:

1. A multi-channel cluster-based TDMA MAC protocol to coordinate intra-cluster communications (TC-MAC)
2. A CH election and cluster formation algorithm based on the traffic flow and a cluster maintenance algorithm that benefits from our cluster formation algorithm

3. A multi-channel cluster-based CDMA/TDMA hybrid MAC protocol to coordinate inter-cluster communications

I will show that TC-MAC provides better performance than the current WAVE standard in terms of safety/update message reliability and non-safety message delivery. Additionally, I will show that my clustering and cluster maintenance protocol provides more stable clusters, which will reduce the overhead of clusterhead election and re-clustering and leads to an efficient hierarchical network topology.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

First and foremost, I would like to express my sincere gratitude to my advisor Dr. Michele C. Weigle for the continuous support of my Ph.D. study and research, for her patience, motivation, enthusiasm, and immense knowledge. Her guidance helped me in all the time of research and writing of this dissertation. I could not have imagined having a better advisor and mentor for my Ph.D study.

My sincere thanks also goes to Dr. Hussein Abdel-Wahab, former GPD of the Department of Computer Science, for the support, motivation and guidance. Thank you for believing in me from the first day I joined the program.

Special thanks to Dr. Stephan Olariu for being a great mentor and for teaching me how to be a researcher. I will never forget when you always welcomed me in your office, I enjoyed every discussion and conversation we had.

Also, I would like to thank the rest of my dissertation committee: Dr. Kurt Maly and Dr. Dimitreie C. Popescu for their encouragement, and insightful comments.

My colleagues in the Wireless Network Research Lab and the staff of the Department of Computer Science, thanks for you help and support.

My uncle, Abdel-Aziz Almalag, thank you for being there for me, my mother, my brother, and my sisters. I do remember the first day when I went to school, it was with you. I remember you showing up in my school and asking my teachers about me.

I would like to thank my brother Humood, and my sisters Naema and Nawal, for supporting me throughout my life, and taking care of my mother while I am away from home.

I cannot thank my mother, Modi AlOmair, enough. She always had confidence in me and offered me encouragement and support in all my endeavours.

Last but not the least, to my dear father, God bless his soul, that I have never seen. You are always in my heart wherever I go.

TABLE OF CONTENTS

Page

Chapter

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

According to the World Health Organization (WHO) [6], approximately 1.3 million people die each year on the world's roads and between 20 and 50 million sustain non-fatal injuries. Road traffic injuries are the leading cause of death among young people, aged between 15 and 29. Many accidents may be avoided by having vehicles communicating with each other to exchange messages to warn the drivers about unsafe situations on the road. Moreover, the Texas Transportation Institute [7] reported that in 2009 the cost of traffic congestion in the US was about $115 billion. This cost based on the wasted time and fuel. The total hours wasted in traffic congestion in the US alone is about 4.8 billion hours, and about 3.9 billion gallons of fuel is wasted. Besides the economic cost, traffic congestion leads to more pollution in our cities.

Vehicular Ad Hoc networks (VANETs) are an important component of Intelligent Transportation Systems (ITS) [8], which apply information technologies in vehicles and transportation infrastructure. VANETs enable the exchange of messages between vehicles and between vehicles and infrastructure, as shown in Figure 1. Such communications aim to increase safety on the road, improve transportation efficiency and provide comfort to drivers and passengers.

In the US, VANETs use 75 MHz of spectrum in the range of 5.850 to 5.925 GHz specially allocated by the U.S. Federal Communications Commission for Vehicle-to-Vehicle communication (V2V) and Vehicle-to-Infrastructure communication (V2I) using Dedicated Short Range Communication (DSRC) technology [9]. The spectrum band is divided into seven 10 MHz channels (Figure 2). Channel 178 is the control channel (CCH), which is used for beacon messages, event-driven emergency messages, and service advertisements. The other six channels are service channels (SCH) to support non-safety messages.

The IEEE has developed the 1609 family of standards for Wireless Access in Vehicular Environments (WAVE) [10]. In WAVE, the IEEE 1609.4 trial standard [2] operates on top of IEEE 802.11p in the MAC layer. IEEE 1609.4 focuses on multi-channel operations of a DSRC radio. There is a sync interval (SI) of the length of

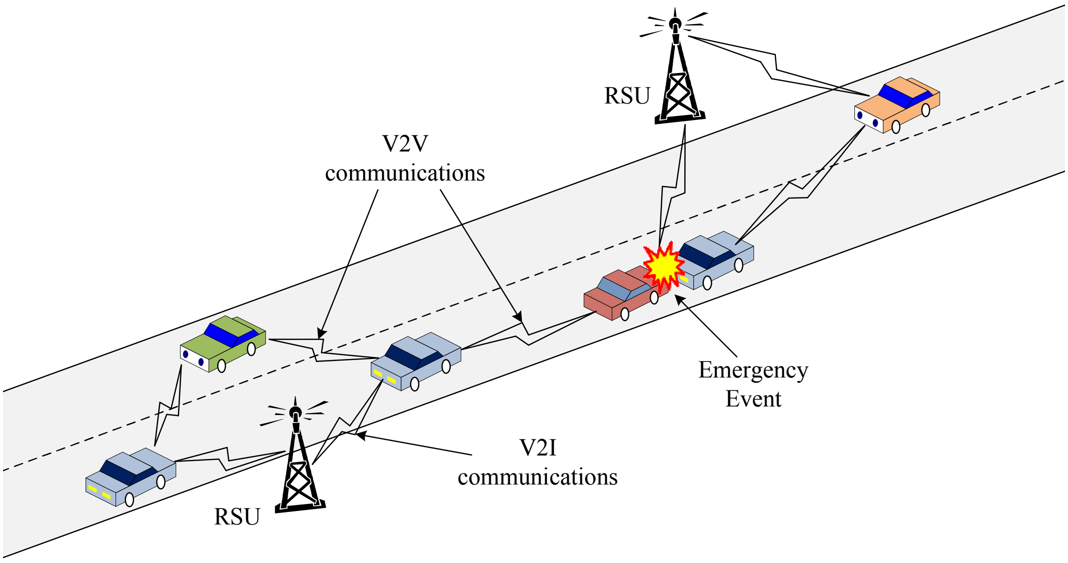


Fig. 1. An example of VANETs

100 msec that consists of a CCH interval (CCHI) and a SCH interval (SCHI), each separated by a guard interval, as shown in Figure 3. All radio devices are assumed to be time-synchronized using Global Positioning System (GPS). During the CCHI, all radios must be tuned to the CCH to broadcast updates and listen for messages from neighbors and road-side units (RSUs). During the SCHI, vehicles may tune to the SCH of their choice depending on the services offered. The reason for having the length of the SI equal to 100 msec is that update messages from vehicles need to be broadcasted at least once every 100 msec [11].

Several ongoing research projects supported by car manufacturers, governments and academia, are establishing standards for VANETs, obtaining frequency spectrum allocations, implementing protocols and applications, and running field trials. However, the widespread deployment of such technology poses several technical issues, concerning architecture, routing, mobility, channel modeling, security, performance, and applications definitions.

1.1 MOTIVATION

Although the primary purpose of VANETs is to increase safety on the roads by running several safety applications, e.g., cooperative collision warning, VANETs can also provide several non-safety applications, from notifications of traffic conditions

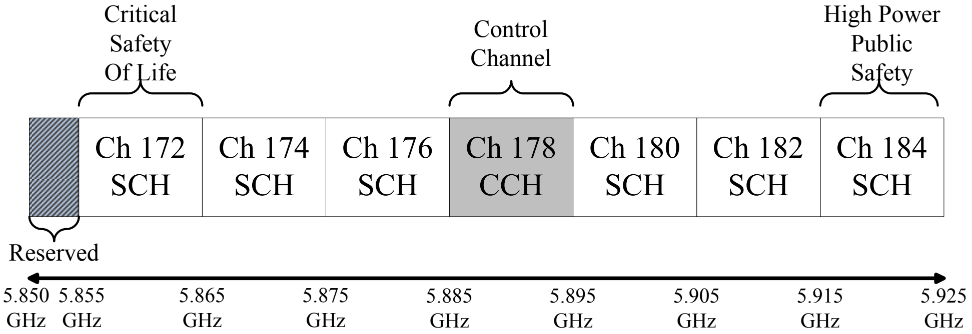


Fig. 2. US DSRC spectrum allocation

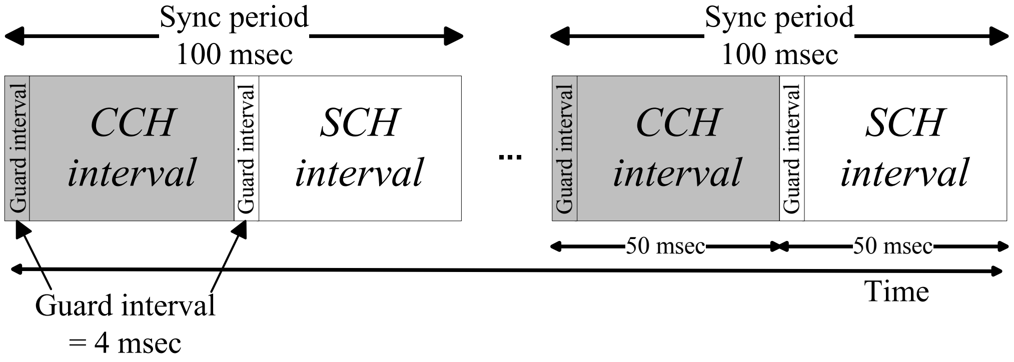


Fig. 3. Division of time into CCH intervals and SCH intervals, IEEE 1609.4 standard

to file sharing. Unfortunately, it has been shown that using WAVE VANETs cannot support both safety and non-safety applications with high reliability at high traffic densities. Either safety applications or non-safety applications must be compromised. To maintain the 100 *msec* requirement of safety applications and ensure reliability, the CCHI must be lengthened and the SCHI shortened [12].

As an ad-hoc network, a VANET cannot rely on specialized infrastructure, such as Access Point, to support network stability. Each node in a VANET is required to maintain its own connectivity to other nodes in the network. With the large number of nodes and the lack of routers, a flat routing scheme, where each node acts as a router, may cause serious scalability and hidden terminal problems. One possible solution to these problems is hierarchical clustering, as illustrated in Figure4. In addition, using clustering can lead to more node coordination and fewer nodes interfering with each other.

A cluster is a group of nodes that can communicate without disconnection and

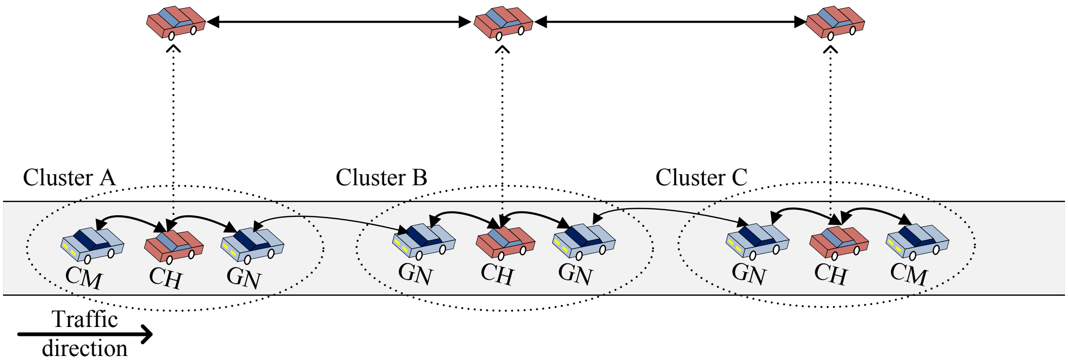


Fig. 4. Clustering envionment

that identify themselves to be part of a cluster. These nodes select a clusterhead *CH* to coordinate the communication among themselves. Clustering in VANETs requires selecting a *CH* that produces a stable cluster. Cluster stability is also affected by the dynamics of the vehicles in the cluster. New vehicles joining the cluster and other vehicles leaving the cluster change the topology of the cluster. Having a simple clustering scheme in forming and maintaining clusters will save a significant amount of time and channel bandwidth needed to complete this process.

Since safety applications of vehicular communication have stringent reliability and delay requirements, giving each vehicle the time to send safety messages without interfering with other vehicles is required. Time Division Multiplexing Access (TDMA) is a technique that can be used to assign unique time slots to each vehicle in the cluster. The goal of any assignment scheme is to make the process of assigning slots easy and straightforward. Also, as important as the safety messages are, non-safety messages need to be delivered even if there are a lot of safety messages.

1.2 OBJECTIVE

The main objective of this work is to design and implement a Medium Access Control (MAC) protocol for V2V and V2I communications for VANETs. This protocol integrates the centralization approach of clusters and a new scheme for slot reservations, where cluster members are assigned local IDs by the *CH*. In this technique, unlike WAVE, all vehicles are able to tune to the Control Channel (CCH) or the Service Channels (SCHs) if needed during the time cycle. It is designed to allow vehicles to send and receive non-safety messages without any impact on the reliability of sending and receiving safety messages even if the traffic density is high.

In this work, I propose a dynamic TDMA slot assignment scheme for cluster-based VANETs. In this scheme, the collision-free intra-cluster communications are managed by the *CH* using local IDs. In addition, I propose a CDMA/TDMA hybrid MAC protocol for inter-cluster communications.

As a result, I encounter three important problems. These problems are cluster formation, cluster maintenance, and intra-cluster and inter-cluster communications. In this work, I propose three algorithms to solve the addressed problems.

1.3 THESIS STATEMENT

TDMA slot reservation in cluster-based VANETs will improve the performance of delivering non-safety messages with little impact on the delivery of safety messages. This performance will be measured by the delivery delay and reception probability of safety and non-safety messages compared to the current WAVE channel switching. I also measure the overhead by counting the extra control messages required for channel assignment and cluster maintenance.

1.4 CONTRIBUTIONS

The overall objective is to allow vehicles to send and receive non-safety messages with little impact on the reliability of sending and receiving safety messages even if the traffic density is high. This is accomplished through three tasks, which are the main contributions of this work:

- *A multi-channel cluster-based TDMA MAC protocol to coordinate intra-cluster communications (TC-MAC).* The proposed protocol can be used for clustering management and communications. This protocol integrates the centralization approach of clusters and a new scheme for slot reservation, using cluster members' local IDs. In this technique, all vehicles are able to tune to the *CCH* or one of the *SCHs* if needed during the time cycle. In other words, the time cycle is not divided into two different intervals, *CCH* Interval and *SCH* Interval as with WAVE. Details will be discussed in Chapet 3.

- *A CH election and cluster formation algorithm based on the traffic flow and a cluster maintenance algorithm that benefits from our cluster formation algorithm.* Rather than considering some of VANET characteristics in the election of the *CH*, my algorithm puts into account the traffic flow on the road. The design and implementation of this *CH* election and cluster formation algorithm shows fewer *CH* changes, which reduces the overhead of re-clustering and delivers an efficient hierarchical network topology. During the cluster formation process, the cluster members will be assigned local IDs by the *CH*. Vehicles in VANETs are allowed to move freely. Therefore, I propose a new cluster maintenance algorithm that handles topology changes caused by mobility changes. The proposed algorithm takes advantage of the local IDs that are assigned in our cluster formation algorithm. Details will be discussed in Chapter 4.

- *A multi-channel cluster-based CDMA/TDMA hybrid MAC protocol to coordinate inter-cluster communications.* I propose a MAC protocol that enables vehicles to communicate with vehicles in different clusters and with *RSUs*. In addition, the hidden and exposed terminal problems are addressed by the proposed protocol. Details will be discussed in Chapter 5.

1.5 OUTLINE

This work is organized as follows:

- Chapter 2 presents a background on VANETs and provides an overview of the IEEE standards for VANETs. It also presents a study of already existing MAC protocols for VANETs.
- Chapter 3 presents the TDMA slot assignment algorithms for intra-cluster communications (TC-MAC).
- Chapter 4 presents the proposed *CH* election and the cluster formation algorithms. It also presents the cluster maintenance algorithm.
- Chapter 5 presents the CDMA/TC-MAC hybrid protocol for the inter-cluster communications.
- Chapter 6 presents an evaluation of the proposed system performance by using extensive simulations. The results are studied and analyzed carefully.
- Chapter 7 presents an application using TC-MAC for peer-to-peer file sharing in VANETs.

- Chapter 8 gives the summary and conclusion of my work.

CHAPTER 2

BACKGROUND AND RELATED WORK

In this chapter, I will outline the characteristics of VANETs, as well as the type of messages in VANETs. Then I will give a background of IEEE standards for MAC protocols for VANETs. I also will explain the Time Division Multiple Access (TDMA) Code-Division Multiple Access (CDMA) as two different techniques for channel partitioning. After that, I will review some *CH* election and cluster maintenance algorithms. In the end, I will review some alternate MAC protocols for VANETs.

2.1 CHARACTERISTICS OF VANETS

The specific characteristics of VANETs make their quantitative and qualitative analysis particularly critical, especially when designing MAC protocols. Even though VANETs are considered to be a class of Mobile Ad Hoc Networks (MANETs), they have a number of specific characteristics that make many solutions for general MANETs unsuitable for VANETs [13]. Some of the VANETs characteristics that influence the design of an ideal MAC protocol are:

- *Number of nodes:* The node density of a VANET may vary. It can be small as in rural areas or large as during rush hour in a large city. It is important to have a MAC protocol that can deal with both cases. The main challenge in rural areas is network disconnection, while scalability is the main challenge in high density areas.

- *High node mobility:* Nodes in a VANET can move at very high speeds (160 km/h), which might lead to frequent disconnection among nodes. If one node is moving at a very high speed (140 km/h) and connected to a node that is moving at a very low speed (30 km/h), the lifetime of the link will be short.

- *Predictable network topology:* The movement of nodes in a VANET is somewhat predictable due to the fact that node movement is constrained by the road topology.

- *Frequently changing network topology:* Due to high node mobility, the network topology in a VANET changes very frequently. It is important to have a MAC protocol that can adapt to frequent changes in the topology in a seamless way.

- *Availability of location information:* Location information can be provided by having a Global Positioning System (GPS) receiver on board. Having such information for communications not only can reduce delivery latency of message dissemination but can increase system throughput.

- *Infrastructure support:* Unlike most MANETs, VANETs can take advantage of infrastructure on the roads. This could enhance the performance of VANET MAC protocols.

- *No power limitation:* Unlike MANET nodes, nodes in VANET have no energy limit. They depend on a good power supply (e.g. vehicle battery). This allows nodes to have better computation resources.

2.2 MESSAGES AND TRANSMISSIONS IN VANETS

Besides the characteristics of VANETs, MAC protocol design should consider different types of messages and their dissemination requirements.

2.2.1 TYPES OF MESSAGES

In VANETs, there are three types of messages: periodic messages, event-driven messages, and informational messages. These three types of messages have different priorities but must share the same bandwidth.

- *Periodic Messages* are generated to inform nearby vehicles about the vehicles current status, for example, speed, position, and direction [14]. Because information in periodic messages is important to all vehicles surrounding the sender, these messages need to be broadcasted frequently. Because of this, periodic messages may cause the broadcast storm problem, leading to contention, packet collisions, and inefficient use of the wireless channel [15].

- *Event-driven Messages* are emergency messages sent to other vehicles based on unsafe situations that have been detected. This type of messages has a very high priority. There are several applications in VANETs that use this type of message, for example, Collision Avoidance Systems (CCA) [16]. The challenge with this type of message is that the sender needs to make sure that all vehicles intended to benefit from these messages receive them correctly and quickly [17].

- *Informational Messages* are non-safety application messages. They help in making driving more convenient and comfortable. An example of this type of message is one facilitating Internet access to the vehicles [18]. Unlike the other type of messages, this type does not require high priority, but may require a high transmission rate.

2.2.2 TRANSMISSIONS

Most of the transmissions in VANETs are broadcast. Broadcast wireless transmissions do not use MAC-layer acknowledgments (ACKs). ACKs are normally sent by a receiver for each frame successfully received. When a node fails to receive an ACK in a certain amount of time, it doubles its CW_{min}, which increases the amount of time it will likely have to wait before sending the retransmission. Since broadcast has no ACKs and therefore no retransmissions, CW_{min} is never adjusted. Because of this, all nodes will have the same CW_{min}, which increases the probability that two nodes will pick the same back-off timer value (BT) value. IEEE 802.11 includes a mechanism (RTS/CTS) to prevent collisions for unicast transmissions, but unfortunately most VANET transmissions are beacons sent via broadcast and cannot use RTS/CTS.

2.3 IEEE STANDARDS FOR MAC PROTOCOLS FOR VANETS

In the US, the Federal Communication Commission (FCC) has allocated 75 MHz of spectrum at 5.9 GHz for Dedicated Short-Range Communications (DSRC) [9], which provides high-speed communication between the vehicles and road-side units (RSUs). DSRC is divided into 7 channels, each 10 MHz wide, as shown in Figure 5. Channel 178 is the control channel (CCH), which is used for beacon messages, event-driven emergency messages, and service advertisements. The remaining six service channels (SCHs) support non-safety applications provided by RSUs. The IEEE has completed the 1609 family of standards for the Wireless Access in Vehicular Environments (WAVE) standard [10] for vehicular communications. In the remainder

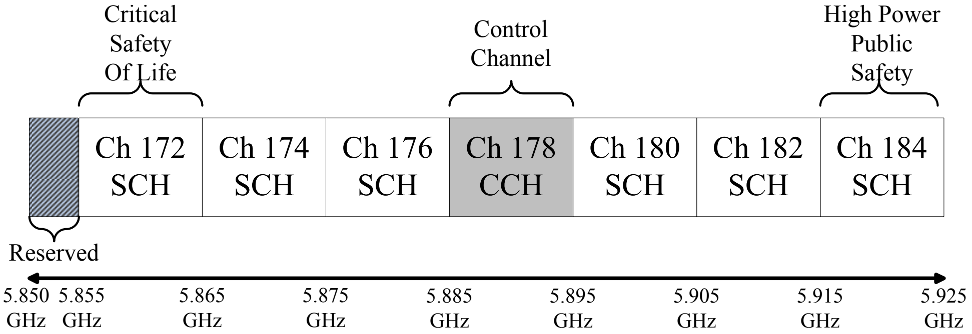


Fig. 5. US DSRC spectrum allocation

of this section, we explain the WAVE standard as well as the challenges and issues of WAVE MAC.

2.3.1 THE IEEE 1609 WAVE STANDARDS

IEEE 1609 WAVE is family of standards for vehicular communication encompassing vehicle-to-vehicle as well as vehicle-to-infrastructure communications [10]. WAVE specifies the following standards, as shown in Figure 6:

- IEEE 1609.1 specifies the services and interfaces of the WAVE Resource Manager application, [19].
- IEEE 1609.2 defines secure message formats and processing [20].
- IEEE 1609.3 presents transport and network layer protocols, including addressing and routing, in support of secure WAVE data exchange [21].
- IEEE 1609.4 specifies MAC and PHY layers [2], which are based on IEEE 802.11. This is the main focus of this work.

2.3.2 IEEE 1609.4 STANDARDS

In WAVE, the IEEE 1609.4 trial standard [2] operates on top of the IEEE 802.11p in the MAC layer. IEEE 1609.4 focuses mainly on dealing with multi-channel operations of DSRC radio, as shown in Figure 7. There is a sync interval (SI) that consists of a CCH interval (CCHI) and a SCH interval (SCHI), each separated by a guard interval, as shown in Figure 8. All radio devices are assumed to be synchronized

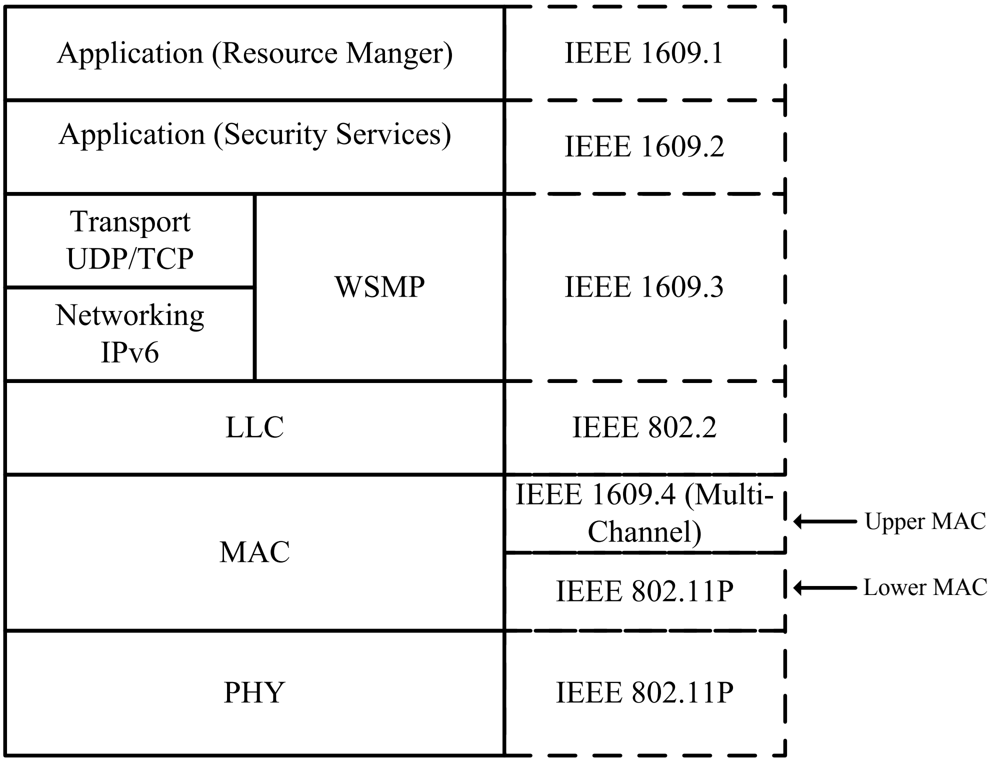


Fig. 6. The WAVE Protocol stack

using Global Positioning System (GPS). During the CCHI, all radios must be tuned to the CCH to broadcast updates and listen for messages from neighbors and RSUs. During the SCHI, vehicles may tune to the SCH of their choice depending on the services offered.

The standard defines the length of the SI as 100 *msec*, based on the desire of having 10 safety messages sent per second. This desire came from the allowable latency requirements of Life-Critical safety applications, which is 100 *msec*. It also defines a Guard Interval (GI) at the start of each CCHI and SCHI. The purpose of the GI is to account for the channel switching. Currently, the value of the GI is from 4 to 6 μsec, which is the time overhead for a radio to be tuned to and made available in another channel.

2.3.3 THE IEEE 802.11P STANDARD

The IEEE 802.11p [3] standard is the foundation of the IEEE 1609 WAVE family of standards. It defines the physical and the medium access control layers. The

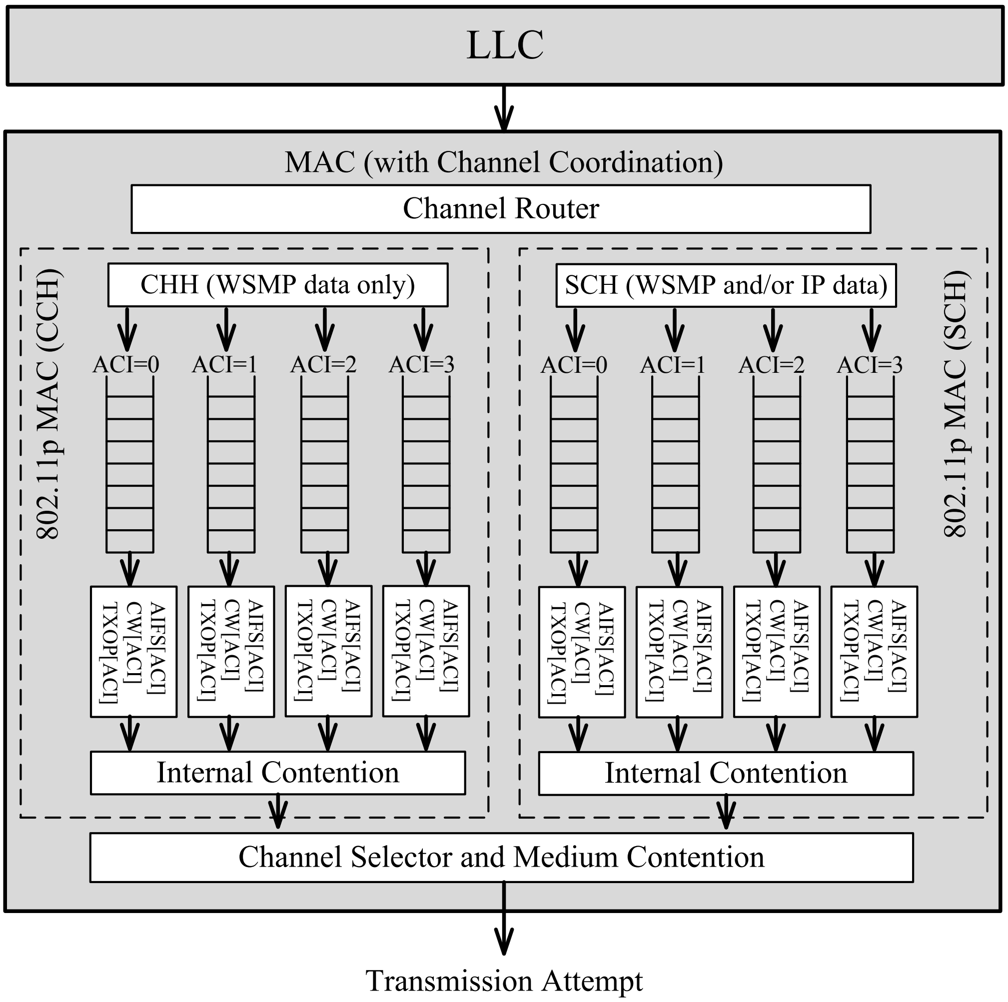


Fig. 7. Reference architecture of the MAC channel coordination. (Figure 4 of IEEE Trail-Use Standard for Wireless Access in Vehicular Environments (WAVE)-Multi-channel Operation)

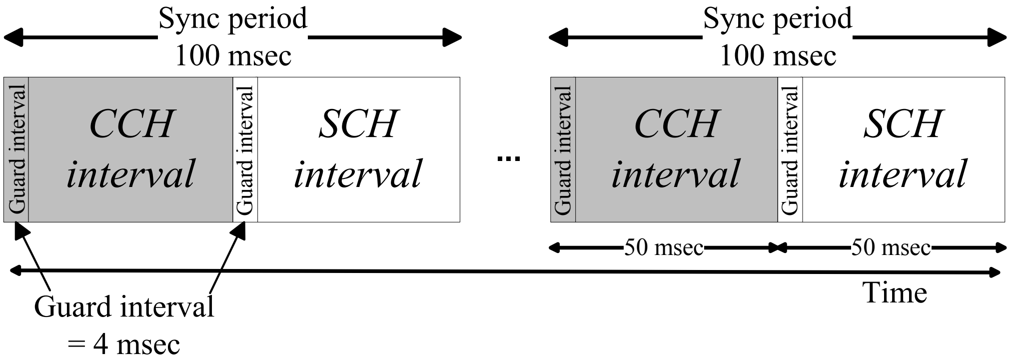


Fig. 8. Division of time into CCH intervals and SCH intervals, IEEE 1609.4 standard (based on [2])

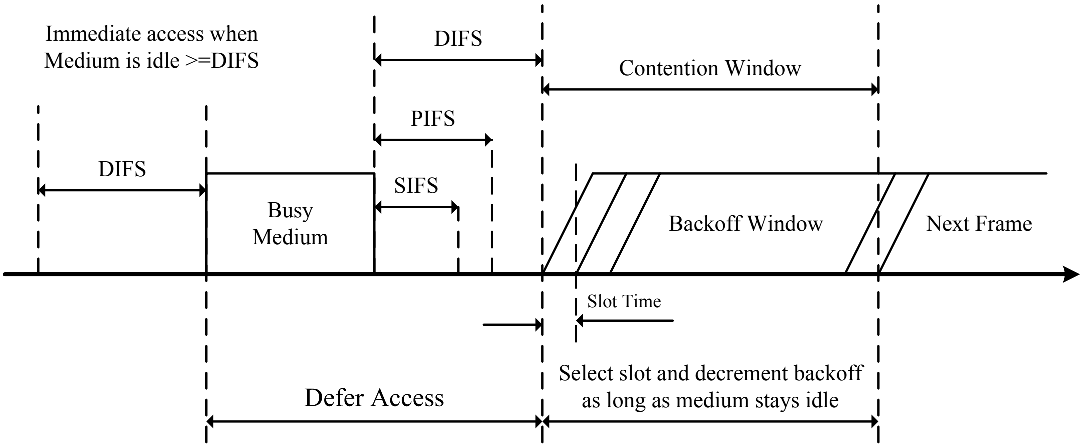


Fig. 9. Distributed coordination function for channel access. (Based on figure from [3])

WAVE stack uses IEEE-802.11p, which is based on CSMA/CA (Carrier Sense Multiple Access/Congestion Avoidance) mechanism to access the medium as defined in IEEE 802.11, see Figure 9. This medium access protocol checks the channel status before transmitting frames from the MAC layer. If the channel is idle and during the a DIFS (DCF Interframe Space) time interval, the frame can be transmitted. On the other hand, if the channel is busy, or becomes busy during the DIFS time interval, the transmission is deferred using the backoff mechanism. The purpose of the backoff mechanism is to avoid a collision with other node that is currently transmitting on the same medium. IEEE-802.11p includes the QoS amendments of IEEE-802.11e. Recently, IEEE has completed work on the 802.11p, local area network standard that employs IEEE 802.11e Enhanced Distributed Channel Access (EDCA). Figure 7 gives an overview of the EDCA architecture and the type of channels that are supported, CCH and SCHs. For the IEEE 802.11p, different Arbitration Inter Frame Space (AIFS) and Contention Window (CW) values are chosen for different application categories (ACs). There are four available data traffic categories with different priorities: background traffic (BK), best effort traffic (BE), voice traffic (VO) and video traffic (VI). Each data traffic category has its own queue; four different queues for each channel. Table 1 shows the parameter settings for different application categories in IEEE 802.11p .

Based on the nature of VANET, IEEE 802.11p has to have different MAC operations than IEEE 802.11. Here is a brief description of some of the changes at IEEE

TABLE 1. IEEE 802.11p parameter settings for different applications categories

AC	CWmin	CWmax	AIFSN
VI	3	7	2
VO	3	7	3
BE	7	225	6
BK	15	1023	9

802.11 MAC [22]:

- WAVE mode: Since safety communications in VANETs demand fast data exchange, IEEE 802.11 MAC operations are too time-consuming. Scanning channels for the beacon of a Basic Service Set (BSS) and performing multiple handshakes to establish the communications is not affordable. Therefore, in the WAVE mode, vehicles are in the same channel and the same BSSID in order to communicate without any additional overhead.
- WAVE BSS: The WAVE standard defines a new BSS type, WAVE BSS (WBSS). When a vehicle/RSU wants to form a WBSS, it transmits an on-demand beacon. This beacon is of a specific format and used to advertise a WAVE BSS. The process taken to join the WBSS or not is done by the upper layers. Also, the WAVE advertisement includes all the information needed by the receiver to configure itself into a member of the WBSS. The way WBSS works leads to low setup overhead by discarding all association and authentication processes.

2.3.4 CHALLENGES AND ISSUES OF WAVE MAC

As currently envisioned, WAVE allows for the communications of safety and non-safety applications through a single DSRC radio. Unfortunately, it has been shown that DSRC cannot support both safety and non-safety applications with high reliability at high traffic densities. Either safety applications or non-safety applications must be compromised. To maintain the 100 msec requirement of safety applications and ensure reliability, the CCHI must be lengthened and the SCHI shortened. Wang and Hassan [12] studied this scenario, requiring 90% and 95% reliability for CCH messages with different traffic densities. Their results indicate that as traffic density increases, ensuring CCH reliability requires compromising SCH throughput. At high densities, to avoid compromising non-safety applications, the SI would need to be lengthened. This would result in fewer beacon messages sent per second, compromising safety.

2.4 TDMA AND CDMA TECHNIQUES

In this section, I will give a brief description of TDMA and CDMA techniques. First, I will describe TDMA and its main advantage. Then I will describe CDMA.

2.4.1 TDMA

Time Division Multiple Access (TDMA) is a technique used to enable multiple nodes to transmit on the same frequency channel. It divides the signal into different time *frames*. Each frame is divided into several time *slots*, where each node is assigned to a time slot to transmit. The length of the time slot may vary, based on the needs of the node assigned to it. For example, if node i needs to transmit on the channel, it will use its own time slot to transmit. The nodes will transmit in rapid succession, each using its own time slot, as shown in Figure 10.

The main advantage of TDMA is reducing interference between nodes. However, it adds slot allocation complexity.

2.4.2 CDMA

Code Division Multiple Access (CDMA) is a spread spectrum multiple access technique. A spread spectrum technique spreads the bandwidth of the data uniformly for the same transmitted power. A spreading code is a pseudo-random code that has a narrow Ambiguity function, unlike other narrow pulse codes. In CDMA a locally generated code runs at a much higher rate than the data to be transmitted. Data for transmission is combined via bitwise XOR (exclusive OR) with the faster code.

One of the advantages of CDMA is that it can achieve much higher channel bandwidth efficiency for a given spectrum allocation. It also overcomes strong intentional interference.

2.5 CLUSTERING

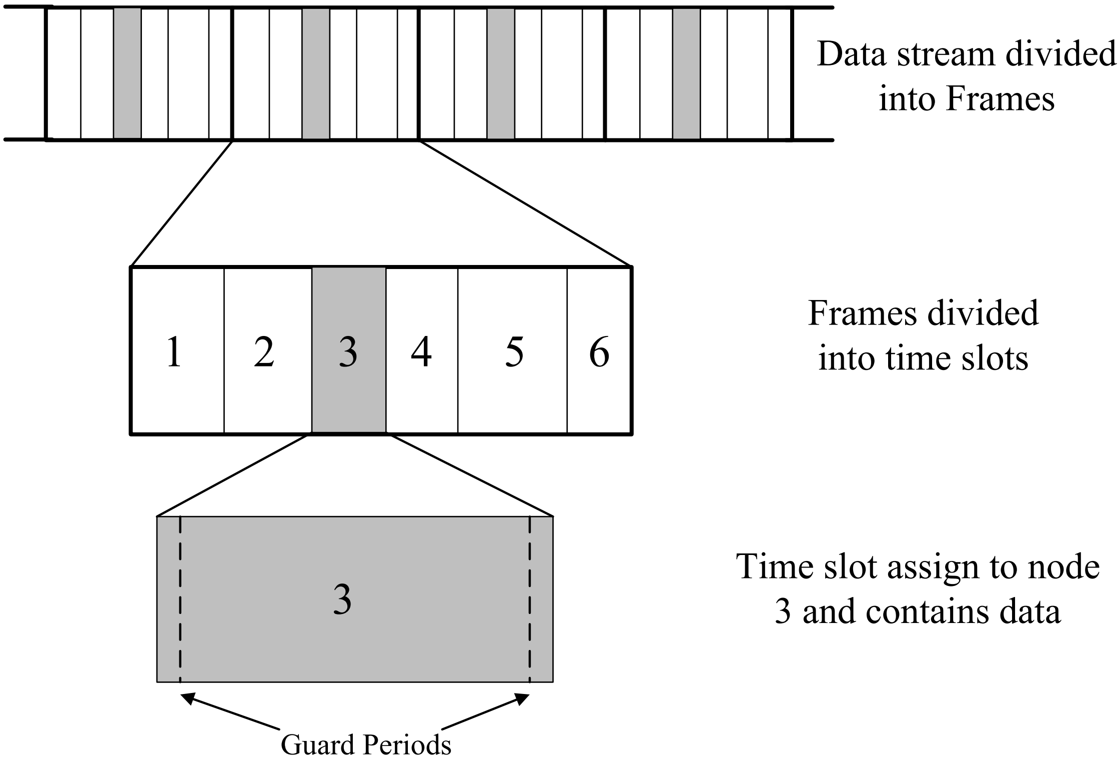


Fig. 10. TDMA frame structure showing a data stream divided into frames and the frames divided into time slots, assuming that we have 6 nodes in the TDMA frame

Clustering is the process of separating the overall network into organized partitions called *clusters*. The *cluster* is conceptual structure where a group of nodes form sub-networks on the road. Nodes in a cluster are classified into three different types, Clusterhead (*CH*), Cluster Members (*CM*), and Gateway Nodes (*GN*). A *clusterhead*, *CH*, is an elected node that is responsible for establishing and organizing the cluster. These responsibility may include routing, relaying and scheduling of intra-cluster communications, and channel assignment for other nodes in the cluster [23]. *Cluster members*, *CM*, are normal nodes that belong to a cluster. *CMs* may participate in routing, when asked by the *CH*. All *CMs* are within one-hop or multi-hop communications range of the *CH*, thus the potential cluster size increases with the transmission range. Gateway Nodes, *GN*, are *CMs* elected by the *CH* to manage communications with adjacent clusters. The *GN* belongs to more than one cluster, acting as a bridge between clusters [24], as shown in Figure 11.

2.5.1 CLUSTER FORMATION IN MANETS

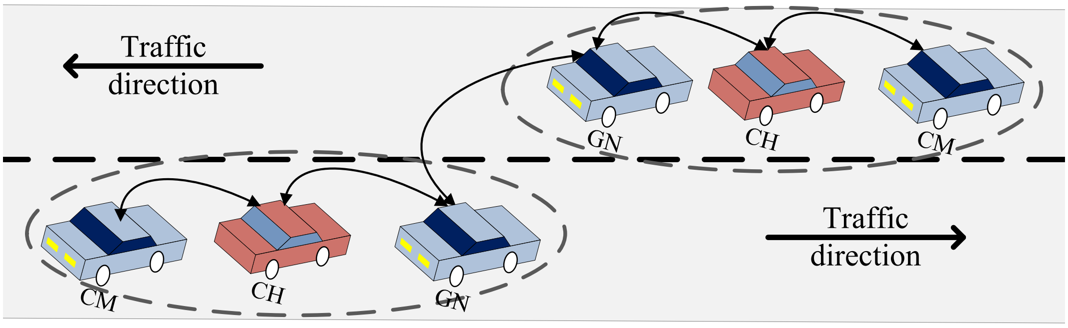


Fig. 11. Two clusters communicating with each other through Gateway nodes (GN) that are assigned by the clusterhead (CH) of each cluster

A Mobile Ad hoc NETwork (MANET) is a self-organizing wireless network of mobile nodes, which can communicate without pre-existing infrastructure. MANETs can have either a flat topology or a hierarchical clustered topology. In large networks, a flat topology faces scalability issues [25]. Routing in MANETs requires flooding to find routes and in large networks this flooding leads to severe congestion. This issue gets more severe as the network is more mobile and the topology changes frequently.

One of the solutions for the scalability issue in MANETs is using a cluster-based approach [26]. Clustering provide scalability by creating a backbone network of nodes. It also provides stability for dynamic networks.

There are several algorithms for cluster formation in MANET. The main task of the cluster formation process is to elect a *CH*. One of the clustering algorithms is Lowest-ID. The Lowest-ID clustering algorithm [27] is based on selecting as the *CH* the member with the lowest ID, assuming each node has unique ID. Simply, each node broadcasts its ID to other nodes in range. When a node receives the messages from other nodes, it determines the *CH* as the node with the lowest ID. This algorithm is very simple and stable for general MANET applications.

Another clustering algorithm is Highest-Degree. The Highest-Degree algorithm [27] selects the *CH* based on the node connectivity to the other nodes in the same cluster. Each node knows the number of other nodes in range and then broadcasts this knowledge to the others. The node with the maximum number of neighbours is selected as the *CH*. This algorithm is one of the basic techniques for *CH* formation in MANETs.

2.5.2 CLUSTER FORMATION IN VANETS

Several algorithms have been introduced for cluster formation in VANETs. Some of these algorithms were first developed for MANETs. For example, Lowest-ID algorithm seems to be simple and stable for MANET; however, this algorithm is not always stable for VANET because the movement of the vehicles is not considered. The Highest-degree algorithm is also not ideal for VANETs. It is not stable for VANETs due to the nature of the nodes movement. If the *CH* changes its behaviour at any moment, the connectivity level could change dramatically.

To have a good VANET clustering algorithm, we have to consider the characteristics of VANET. The Utility Function algorithm [28] is a VANET clustering algorithm that performs better than the previous two algorithms, Lowest-ID and Highest-Degree. This algorithm is based on a multiple-metric weighting algorithm, considering speed, velocity and position. In the process of the *CH* selection, the closest position to the average and the closest velocity to the average of all proximal vehicles are calculated along with connectivity level to determine the most stable *CH*. Periodically, each node broadcasts its status to other nodes in range. When the node receives this information, it starts to evaluate each node by using the utility function. The node with the highest value is chosen to be the *CH*. In a highway environment, this algorithm has been shown to provide better results than the classic MANET algorithms. It puts the position and velocity, which are major VANET characteristics, into consideration. However, it still ignores the traffic flow on the road. For example, in an urban scenario where are many intersections, if the *CH* is located on the leftmost lane, it has to turn left even if most of the vehicles are going straight. In this case, the vehicles will need to perform the process of *CH* selection again.

Rawshdeh and Mahmud [29] proposed grouping vehicles based on the mobility patterns. Vehicles with close speeds will be grouped together in one cluster. But this might lead to having clusters overlap.

We introduce a new cluster formation algorithm. This scheme aims to extend the life of the *CH*. We take advantage of knowing the exact lane of vehicles on the road and then broadcast this knowledge to other nearby vehicles to determine the optimal *CH*. Our method of selecting the *CH* is the key to achieving a more stable cluster.

2.5.3 CLUSTER MAINTENANCE ALGORITHMS

Based on the dynamic nature of vehicles in VANETs and the changes in the network topology, clusters must be updated frequently to maintain the stability of the network. Cluster maintenance is a very important process in any clustering algorithm. It will be performed more often than cluster formation. The cluster maintenance process should be done in a manner that it does not add much communication overhead.

A significant amount of research on cluster maintenance has been done in MANETs [30, 31]. VANETs, however, have different mobility characteristics than MANET. So, applying MANET algorithms in VANET is not always successful. Venkataraman et al. [30] proposed a clustering algorithm that performs formation and maintenance in MANET. In the process of maintenance, if the *CH* is leaving the cluster, the *CH* selection process will be performed again. This will consume a lot of time and channel bandwidth. It is better to do the *CH* changes in a way that does not require a lot communications.

We introduce a new cluster maintenance algorithm that solves the issues of the network topology changes. It addresses such as new cluster member(s) joining, current cluster member(s) leaving the cluster, clusters merging, and *CH* changing. All the problems mentioned are going to be solved with a low amount of overhead. We aim to design an algorithm that makes changes in the topology unseen by most of the cluster members. For example, when the *CH* is about to leave the cluster, it will find a stable *CH* candidate. Then, the current *CH* will switch local IDs with *CH* candidate. After switching, a new more stable *CH* will take over with the need of performing the *CH* selection process again.

2.5.4 CLUSTER-BASED MAC PROTOCOLS

A significant amount of research has been spent in developing new cluster-based MAC protocols, [26, 32, 33, 34, 35, 36, 37]. Gunter et al. [26] proposed schemes where the *CH* takes on a managerial role and facilitates intra-cluster communication by providing a TDMA schedule to its *CMs*. Based on the amount of data the *CMs* have to send, the *CH* assigns a bandwidth and time slots to the *CMs* in each TDMA frame. Su and Zhang [37] proposed a scheme where adjacent clusters are assigned different CDMA codes to avoid interference between clusters. This work shows a substantial reduction in probability of message delivery failure, when compared to the traditional 802.11 MAC. The disadvantages of this work are that it uses two transceivers. It also reserves channels for specific tasks; so if there is no activity on these channels, the channels will be wasted.

Much of the recent VANET research discussing cluster-based MACs and routing schemes also present a low-maintenance clustering algorithm. Each of these algorithms works essentially the same way, whereby nodes periodically transmit HELLO beacons to indicate their present state. States can be one of the following: Undecided, *CH*, Cluster Member, and sometimes Gateway. An undecided node will join the first *CH* that it hears a HELLO beacon from (or joins all *CH* if Gateway nodes are allowed). If the node does not hear from a *CH* within a given time period, it will become a *CH* itself. In addition, protocols are introduced to deal with colliding clusters, which occurs when two *CH* come within range of one another. During a cluster collision, one *CH* decides to give up its status to the other. This technique is used by Su and Zhang [37] without regard for mobility. Gunter et al. [26] proposed a scheme where mobility is addressed during cluster collision, whereby the winning *CH* is the one with both lower relative mobility and closer proximity to its members. Alternatively, Kayis and Acarman [36] address mobility by first classifying nodes into speed groups, such that nodes will only join a *CH* of similar velocity.

2.6 ALTERNATE MAC PROTOCOLS FOR VANET

The issues of MAC protocols for WAVE, as described above, led researchers in developing new MAC protocols for VANETs. In general, MAC protocols can be classified into three different categories: channel partitioning, random access, and taking turns [38]. In this section, we survey some of the most recent research efforts on MAC protocols for VANET. We will discuss the MAC protocols based on the categorization above.

2.6.1 CHANNEL PARTITIONING

Channel partitioning MAC protocols are based on sharing the channel efficiently at high uniform load. In MAC layer, channel partitioning is done using the following methods: Time-Division Multiple Access (TDMA), Frequency-Division Multiple Access (FDMA) and Code-Division Multiple Access (CDMA). In this section, we will discuss some of the MAC protocols for VANET designed using TDMA.

In VANETs, TDMA is used to enable multiple nodes to transmit on the same frequency channel. It divides the signal into different time frames. Each time frame is divided into several time slots, where each node is assigned to a time slot to transmit [39]. The length of the time slot may vary, based on the needs of the node assigned to it. The nodes will transmit in rapid succession, each using its own time slot.

The main advantages of protocols developed under this category are reducing interference between nodes and providing fairness. However, they add allocation complexity and suffer from inefficient channel utilization at low loads.

Yu and Biswas [4] proposed Vehicular Self-Organized MAC (VeSOMAC), a MAC protocol for inter-vehicular wireless networking using DSRC. They designed a self-configuring TDMA slot reservation protocol capable of inter-vehicle message delivery with short and deterministic delay bounds. To achieve the shortest delay, vehicles determine their TDMA time slot based on their location and movement on the road. Also, the TDMA slot assignment is designed to be in the same sequential order with respect to the vehicles physical location.

As shown in Figure 12, if vehicle 1 detects an emergency event that needs to be disseminated to other vehicles behind it, the message will go from vehicle 1 to vehicle 5 through vehicles 2-4; assuming that each vehicle is in range of only one vehicle ahead and one vehicle behind. Also there is an assumption that as soon as the message is transmitted, it can be sent by the next vehicle without processing or propagation delay. If the TDMA slot assignment is not based on the physical location of the vehicle in the platoon 1-2-3-4-5, it may take more than one TDMA frame for the emergency message to reach vehicle 5. For example, we show an alternate assignment of 4-3-2-1-5 as shown in Figure 13, vehicle 1 is assigned to a time slot that is after the time slot assigned to vehicle 2. That means vehicle 2 will finish sending to its neighbors using its time slot before it hears the message from vehicle 1 in time frame 1. The same case applies when vehicle 2 tries to send to vehicle 3. We notice that in order for the message to be delivered from vehicle 1 to vehicle 5, four time frames are needed. Using the VeSOMAC protocol will minimize delivering the message from vehicle 1 to vehicle 5 to only one time frame by using vehicle location for the time slot assignment, as shown in Figure 14.

To solve the direct and hidden terminal collisions in VANET, VeSOMAC needs to satisfy timing constraints where no two one-hop or two-hop neighbors slots can overlap. It also uses an in-band header bitmap to exchange slot allocation information among vehicles. To achieve faster message delivery, VeSOMAC uses an ordering constraint where the vehicle ahead will be assigned to an earlier time slot than the

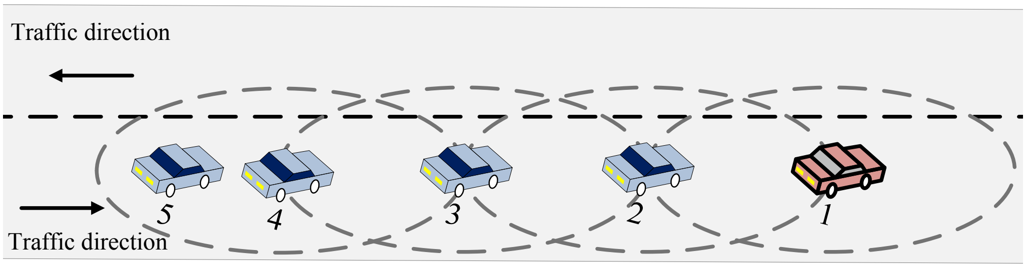


Fig. 12. Highway scenario where the first vehicle needs to disseminate an emergency message. (Based on figure from [4])

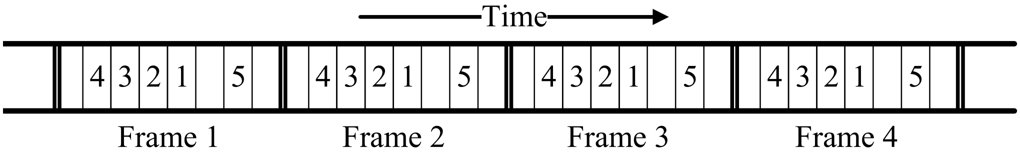


Fig. 13. TDMA slot assignment without using VeSOMAC, regular TDMA

vehicle behind it in the platoon.

In this protocol, the process of assigning time slots is done without using infrastructure or virtual schedulers such as a leader vehicle. However, the assumption of forwarding messages without processing time or propagation delay is unrealistic. It shows that if the message needs to be delivered from the tail to the head of the platoon, it will need a time frame for each hop. So far, VeSOMAC does not explain the communication between vehicles and RSUs.

Omar et al. [5] proposed a multichannel MAC protocol for VANETs, called VeMAC, to reduce interference between vehicles and reduce transmission collisions caused by vehicle mobility. VeMAC is based on a TDMA scheme for inter-vehicle communication. Vehicles in both directions and RSUs are assigned to time slots in the same TDMA time frame. Also, VeMAC is designed based on having one control

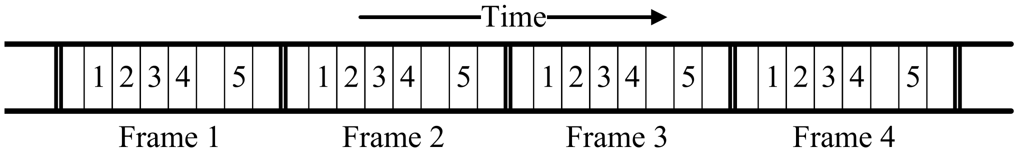


Fig. 14. TDMA slot assignment with VeSOMAC

channel and multiple service channels in the network (as with DSRC/WAVE).

VeMAC assumes that there are two transceivers on each vehicle and that all vehicles are time synchronized using GPS. The first transceiver is assigned to the control channel, while the second transceiver is assigned to the service channels. Vehicles will use the control channel to transmit two types of messages: high priority messages (such as safety messages) and control messages for slot assignment. Since VeMAC considers vehicles in opposite directions, vehicles are said to be either travelling in the right (R) or left (L) direction. With the information provided by GPS, vehicles can determine their direction; if a vehicle is moving from west to east (north to south), it is in the right direction (R) and opposite vehicles are in the left direction (L), as shown in Figure 15. The time frame in VeMAC is divided into three different slots sets, L, R, and F, as shown in Figure 16. Vehicles in the right direction (R) will be assigned to time slots in the time frame from the R slot set, vehicles in the left direction (L) will be assigned to time slots from the L slot set, while RSUs will use slots in the F slot set.

In VeMAC, each vehicle is guaranteed to access the control channel once per frame. Also, vehicles have equal opportunities to announce for services provided on the service channels. To avoid the hidden terminal problem, each vehicle in VeMAC includes in the header of each packet transmitted on the CCH the following information: the time slots used by the vehicle on the SCH, the time slot used by each neighboring vehicle on the CCH, the time slots used by each neighboring vehicle on the SCH, and the position and the current direction of the vehicle. By using this information, each vehicle can determine the set of time slots used by other vehicles within its two-hop range, which will help on avoiding the hidden terminal problem.

2.6.2 RANDOM ACCESS

Random access MAC protocols, also known as contention based protocols, are based on the notion of CSMA. The goal of MAC protocols is to increase throughput, so protocols under this category aim to keep packet collisions to a minimum. The advantage of random access protocols is that they are not sensitive to underlying mobility and topology changes. So, vehicle movement does not impose any reconfiguration overhead due to the network topology changes. Also, CSMA protocols are efficient in low load scenarios. However, in networks such as VANET, the hidden

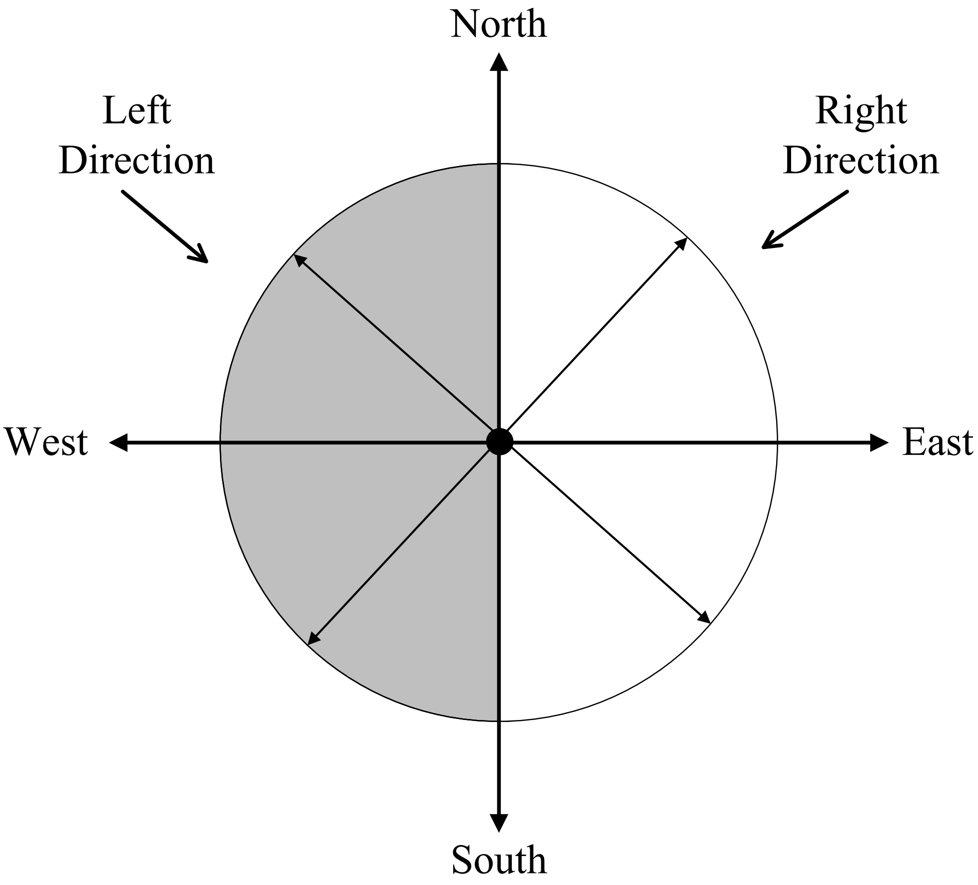


Fig. 15. Vehicles directions in VeMAC. Vehicles in the dark area are Left direction, while others are Right direction

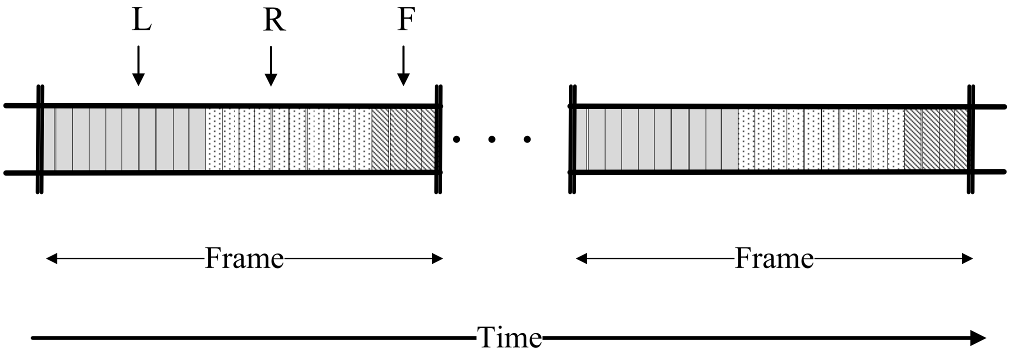


Fig. 16. TDMA time frame in VeMAC shows L, R and F sets. (Based on figure from [5])

TABLE 2. Priority scheme with 4 levels (Based on table from [1])

Priority Level	Vehicle Range	Message Type
Level 0	Far	General
Level 1	Medium	General
Level 2	Low	General
Level 3	Close	Emergency

terminal problem and exposed terminals affect the system performance. Several random access MAC protocols for VANETs have been proposed, some of which will be described below.

Yang et al. [1] proposed carrier sense multiple access with priority and polling (PP-CSMA) as a MAC protocol for VANETs that is based on a priority scheme in CSMA using different backoff time spacing (BTS). The authors claim that their protocol will provide high priority messages with fast access to the medium.

PP-CSMA proposes the prioritization scheme as a combination of the closeness of the transmitting vehicle to the receiving vehicle and the message type. The position of the transmitting vehicle to the receiving vehicle will determine the vehicle range (far, medium, low, and close); if the range is short, the priority gets higher. Also, the type of message (emergency or general) will have an effect on the priority; emergency messages have higher priority than general messages. Table 2 shows four different levels of priority, the high priority level backs off for the least amount of time.

Besides the priority scheme, the PP-CSMA protocol implements a polling scheme in which the receiving vehicle polls only vehicles with the highest priority level available. Each vehicle maintains a polling table that holds information about other vehicles positions. If a vehicle has an emergency message to be sent, it generates a tone, which is out of the frequency band used for data transmission. If the vehicle is in the receivers polling table, the receiver will clear for the sender to transmit the message. If the polling vehicle does not generate a tone, the receiver vehicle will know it is not an emergency message. The PP-CSMA protocol guarantees that the highest priority messages will always have access to the medium faster than the low priority messages. However, the authors did not mention broadcasting, which is an important challenge in VANET.

In a different work, Suthaputchakun and Ganz [40] proposed a MAC protocol for

TABLE 3. Different message priorities with parameters (Based on table from [40])

Priority Level	Type	Example	CWmin	CWmax	AIFS	Num. of Repet.
Level 1	Accident	Air bag sensor	CWmin/4	CWmin/2	2	3
Level 2	Possibility of Accident	Thermal sensor	CWmin/2	CWmin	3	1
Level 3	Warning	Surface condition	CWmin	CWmax	3	1
Level 4	General	Traffic report	CWmin	CWmax	7	1

VANETs that is based on different message priorities, as in IEEE 802.11e EDCA MAC protocol, with a repetitive transmission mechanism. This protocol aims to increase the communication reliability by using the appropriate number of repetitions per priority class.

Since most of the communications in VANET are broadcast messages with no RTS/CTS or acknowledgment, the network reliability can be low. To solve this problem, the authors proposed a mechanism of retransmission the messages based on the priority of the message. Table 3 shows the priority levels as well as the number of repetitions for each level.

Jiang et al. [41] proposed a set of protocols for safety communications in VANETs. They define three different protocols: CCH congestion control protocol, broadcast performance enhancement protocol, and concurrent multichannel operation protocol. These protocols are designed to address the issues of the current standard and meet the requirements of safety communications in VANETs.

The CCH congestion control protocol is designed around adjusting the generation rate of routine safety (periodic) messages and the transmission power. Based on the communication density [42], vehicles should be able to calculate the generation rate and transmission power of routine safety messages which will maintain a reasonable CCH load. These adjustments are done by each vehicle individually. Each vehicle will listen and understand the targeted channel usage, and then ensure that its share of the channel will keep a reasonable channel congestion level.

For the broadcast performance enhancement, the authors proposed a mechanism that aims to ensure the best possible reception rate for the event safety (event-driven) messages. The way it works is by the sending vehicle collecting feedback from other vehicles on its recent safety message. This feedback will help the safety application(s) on retransmitting the safety message, if needed. The feedback from other vehicles is provided by Piggyback some acknowledgements in their safety messages [41]. For the acknowledgements, vehicles will include the following information in each outgoing safety message: senders position, the intended range of reception, a randomly generated message ID, IDs of most recently received messages, and the reception time of the earliest message in the acknowledgement list.

In the concurrent multichannel operation protocol, the authors intend to increase the level of SCHs utilization, for non-safety messages, with satisfying the safety messages requirements. In VANETs, channel switching between CCH and SCHs is operated every 100 *msec*. Vehicles will operate the switching in order to listen to safety and non-safety messages; if the number of safety messages is high, non-safety messages will have less time to be transmitted. To increase the SCHs utilization, this protocol is built on the concept that listening to all safety messages is not required if: (1) if routine safety messages from all nearby vehicles are heard every few seconds, and (2) all event safety messages from nearby vehicles are received without excessive delay. To do that, the authors used Peercast. The Peercast concept relies on trusting peer vehicles description of recent control channel messaging activities. The following steps will describe the Peercast concept: (1) each vehicle must switch to the CCH every time it has a safety message to transmit, (2) each vehicle must switch to the CCH (e.g. every 100 msec) to hear a few safety messages from its neighbors, (3) while on the CCH: (a) if it hears no safety messages, it may switch back to SCH, (b) else, if it hears an event safety message, it passes it to safety applications and may switch to SCH, (c) else, if it hears an event safety message with unknown ID, it must stay on CCH to capture the repetition of the message before switching back to SCH. (4) each vehicle must switch to CCH every time a safety application requested, (5) each vehicle must switch to CCH every a few second for a short period of time to reorient itself with other vehicles routine messages.

2.6.3 TAKING TURNS

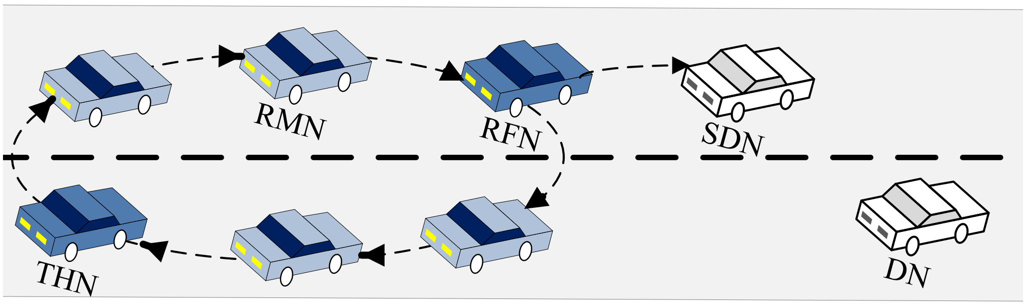


Fig. 17. Token ring in MCTRP with different types of vehicles in the proposed protocol

Taking turns MAC protocols use either polling (master-slave) or token ring techniques. Such techniques provide fairness by giving each node a turn to transmit. They also provide a real time bandwidth allocation. If the node is not transmitting during its turn, the time will be not wasted at the current node. We describe an example of a token-ring based MAC protocol for VANETs in this section.

Bi et al. [43] proposed a multi-channel token ring MAC protocol for inter-vehicle communications in VANET (MCTRP). The protocol aims to reduce the delay of safety messages and improve the dissemination of non-safety messages, based on the multi-channel structure defined in IEEE 802.11p. This can be achieved through adapting multiple rings operating on different service channels.

MCTRP is designed to support more than one token ring at a time. These rings are formed according to the velocity of vehicles and the road conditions. As shown in Figure 17, vehicles forming one ring may have different states: (1) ring founder node (RFN): a node that sets up a ring and has the authority to cancel the ring, adding new nodes to the ring, and deleting nodes from the ring; (2) token holder node (THN): a node that in the ring and holds the token; (3) ring member node (RMN): a node in the ring, but does not hold the token. Vehicles that are not members of a ring may also have different states: (1) semi-dissociative node (SDN): a node that received the joining invitation from the RFN; (2) dissociative node (DN): a node that does not belong to any ring and not in the process of joining any ring.

Vehicles in MCTRP are equipped with two transceivers, I and II. All vehicles in the DN state operate over channel 178 using transceiver I, while other vehicles (non-DN) simultaneously operate over channel 178 using transceiver I and over one of the service channels over transceiver II. All vehicles in the system are time synchronized using GPS.

MCTRP employs three different sub-protocols for resource utilization: a ring coordination protocol, an emergency message exchange protocol, and a data exchange protocol. The ring coordination protocol contains several processes for ring management, such as ring initialization, joining, leaving, ring update, and ring termination. The emergency message exchange protocol is designed to broadcast emergency messages as fast and reliable as possible. That can be done in four steps: (1) when a RMN detects an accident, it transmit an emergency message to the RFN by adopting CSMA CA (with Radio-II during the safety period); (2) a RFN replies with an acknowledgment to the sender RMN, and then broadcasts the message to all other RMNs (with Radio-II for intra-ring notification); (3) at the same time, the RFN broadcasts the message to its neighboring DNs and other RFNs (with Radio-I for inter-ring notification); (4) the other RFN rebroadcast the emergency message to its RMNs. The data exchange protocol is designed on having to data buffers in each node. The intra-ring data buffer (IADB) holds packets to be transmitted to other RMNs in the same ring, and the inter-ring data buffer (IRDB) holds packets to be transmitted to nearby DNs, SDNs and RMNs. For intra-ring data communications, a RMN will send packets when it receives the token, and the IADB is not empty. The transmission time of the THN is controlled by the token maximum hold time T_{MTH}. Once the T_{MTH} is reached, the THN will pass the token to its successor. To ensure token delivery, THN will retransmit the token if it does not receive an acknowledgement (ACK) from its successor and the retransmission timer has expired. If the maximum number of retransmissions is reached with no ACK from the successor, the THN will report to the RFN, and the RFN will delete the successor from the ring and update the ring as well as informing other RMNs. For the inter-ring data communications, data packets are transmitted with CSMA CA mechanism.

MCTRP shows that it can deliver emergency messages in fast way and enhance the network throughput. It also provides fairness among vehicles, in term of channel sharing, and token holding time adjustment.

2.7 SUMMARY

In this chapter, I presented a background about VANET. I explained the characteristics of VANETs, as well as the types of messages in VANETs. I also gave a background about the IEEE standards for MAC protocols for VANETs, including the IEEE 1609 WAVE standards, the IEEE 1609.4 standards, and the IEEE 802.11P standards. After that, I described the challenges and issues of WAVE MAC. For clusters, I explained the main concept of clustering and the cluster formation, including some clustering algorithms. Also in this chapter, I explained the main types of MAC protocols for VANET, including some proposed protocols for each type.

CHAPTER 3

MULTI-CHANNEL CLUSTER-BASED TDMA MAC PROTOCOL FOR INTRA-CLUSTER COMMUNICATIONS (TC-MAC)

In this chapter I will describe our TDMA MAC protocol, TC-MAC. I will start by describing the TDMA slot assignment, and then I will describe the intra-cluster communications. In this protocol, unlike WAVE, all vehicles are able to tune to the CCH or the SCHs if needed during the time cycle. It is designed to allow vehicles to send and receive non-safety messages without any impact on the reliability of sending and receiving safety messages even if the traffic density is high.

3.1 TDMA SLOT ASSIGNMENT

The presentation of my protocol involves several aspects of intra- and inter-cluster communication. In turn, each of these communication regimens is partitioned into cases depending on whether or not the cluster is single-hop. The clustering scheme I am using is clusterhead (*CH*) based, where the consensus is dictated by the *CH*. I also assume that all vehicles are equipped with GPS to ensure that vehicles have synchronized clocks. This protocol is based on the multi-channel DSRC layout, with 1 CCH and 6 SCHs.

To explain my technique, I assume an N-vehicle cluster. The number of vehicles in the cluster must be less or equal to N_{max}; where N_{max} is the maximum number of vehicles in the cluster. The transmission time is partitioned into consecutive, non-overlapping logical TDMA frames. The length of the TDMA frame in TC-MAC is equal to 100 msec. In this case, we guarantee that every vehicle in the cluster sends one update/safety message every 100 msec to meet the safety message requirements. We assume the existence of k slotted SCHs numbered from 0 through k-1[1]. In each SCH, the logical TDMA frames are aligned, i.e. begin and end at the same time.

---

[1] In practice we use k=6 to match DSRC, but we will describe it generally.

Each logical frame contains S number of slots, where $S = \lfloor \frac{N_{max}}{k} \rfloor + 1$ slots. The slots are numbered from 0 through $\lfloor \frac{N_{max}}{k} \rfloor$. All slots are the same size, and the slot size τ is fixed, based on the data rate and the maximum packet size.

We also assume one CCH, channel k, is used by the vehicles and CH for disseminating status and control messages as is done with WAVE. As with the SCHs, the TDMA frame on channel k is divided into slots of size τ. Each time slot on the CCH is divided into k mini-slots used to disseminate status information, such as periodic beacon updates used in safety applications.

By virtue of synchronization, the vehicles know the frame and slot boundaries. The number of vehicles N may change dynamically, and the CH is responsible for updating N and for informing all vehicles in the cluster of the new value of N.

Each vehicle in the cluster will receive a local ID. This local ID is a number from 0 to N. The CH will always have ID 1, and ID 0 is reserved for a virtual vehicle (to be described later). We do not expect all N vehicles in the cluster to be communicating, or active, simultaneously. The CH keeps a list of all the currently-active vehicles and disseminates this list to all the members of the cluster using one of the mechanisms discussed below.

In each logical frame, vehicle j, $(0 \leq j \leq N_{max} - 1)$, owns:

- Channel $(j \ mod \ k)$ during time slot $\lfloor \frac{j}{k} \rfloor$; we also say that vehicle j owns the ordered pair $(j \ mod \ k, \lfloor \frac{j}{k} \rfloor)$
- The mini-slot $(j \ mod \ k)$ of slot $(\lfloor \frac{j}{k} \rfloor - 1)$, on channel k, as illustrated in Figure 18

The basic idea is that in each logical frame, while idle, vehicle j listens to channel $j \ mod \ k$ in slot $\lfloor \frac{j}{k} \rfloor$ and sets the corresponding byte in the CCH in order for other vehicles to be aware of transmitting to vehicle j during j's time slot on the SCH. Notice that the Integer Division Theorem guarantees that if $i \neq j$ then either:

- $\lfloor \frac{i}{k} \rfloor \neq \lfloor \frac{j}{k} \rfloor$ or
- $i \ mod \ k \neq j \ mod \ k$, or both.

This confirms that no two vehicles own the same ordered pair. For an illustration, let N=61 and k=6. Assume we have the network settings in Table 4, and the number

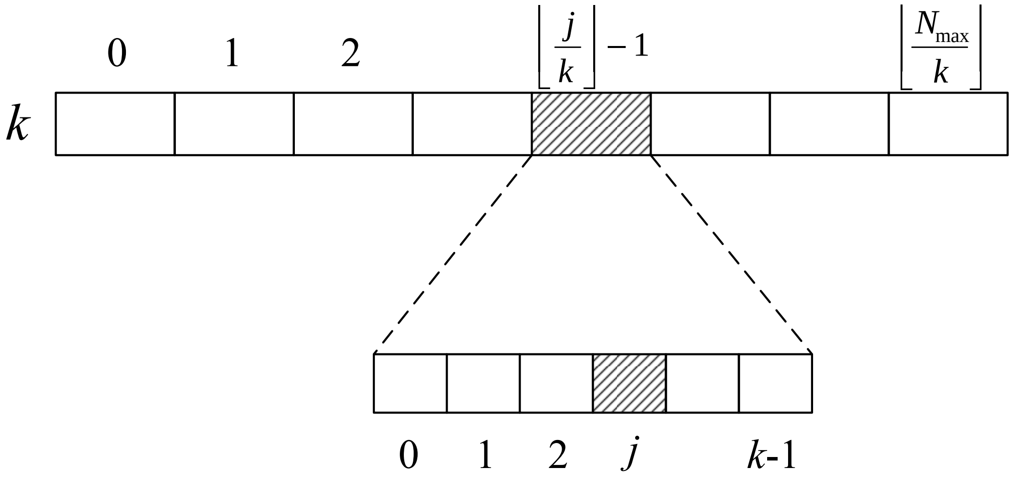
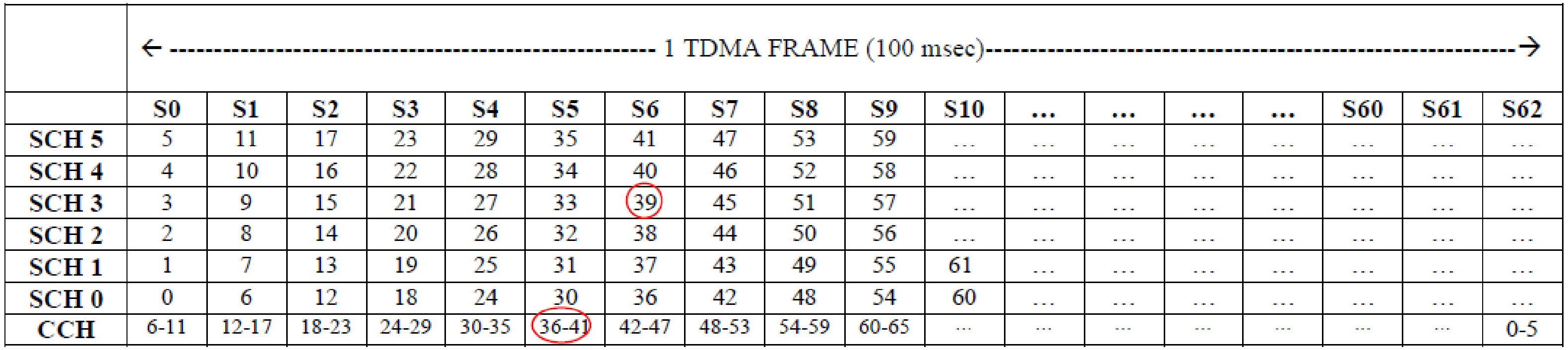


Fig. 18. j's mini-slots on channel k; vehicle j owns a mini-slot on the CCH in the slot preceding its own slot on the SCH

TABLE 4. Network settings

Parameters	Values
Max Safety Packet Size	200 bytes
Max Non-Safety Packet Size	1200 bytes
Data Rate	6 Mbps
Mini Slot Size	0.26 msec
SCH Slot Size	1.6 msec
TDMA Frame Size	100 msec

of TDMA slots on the SCHs will be 65; $N_{max} = 389$. As shown in Figure 19, vehicle with local ID 39 owns channel (39 *mod* 6)=3 during slot $\lfloor \frac{39}{6} \rfloor$=6, as well as 4-th mini-slot on the control channel in slot 6-1=5. We note that for any given N, there are $N_{max} - N$ unused slots in the frame.

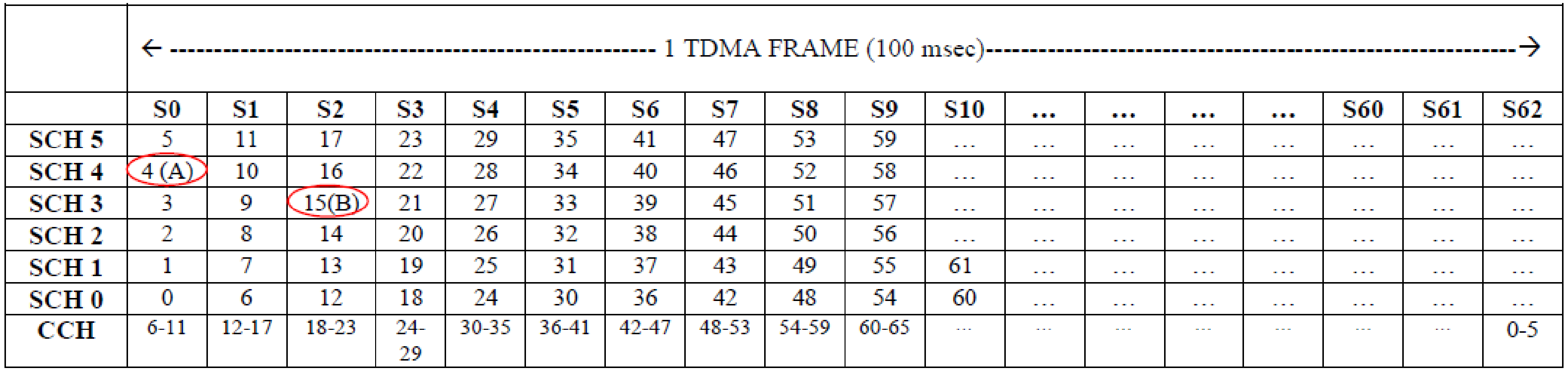


	← -- 1 TDMA FRAME (100 msec)-- →																
	S0	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**	**…**	**…**	**…**	**…**	**S60**	**S61**	**S62**
SCH 5	5	11	17	23	29	35	41	47	53	59	…	…	…	…	…	…	…	…
SCH 4	4	10	16	22	28	34	40	46	52	58	…	…	…	…	…	…	…	…
SCH 3	3	9	15	21	27	33	39	45	51	57	…	…	…	…	…	…	…	…
SCH 2	2	8	14	20	26	32	38	44	50	56	…	…	…	…	…	…	…	…
SCH 1	1	7	13	19	25	31	37	43	49	55	61	…	…	…	…	…	…	…
SCH 0	0	6	12	18	24	30	36	42	48	54	60	…	…	…	…	…	…	…
CCH	6-11	12-17	18-23	24-29	30-35	36-41	42-47	48-53	54-59	60-65	…	…	…	…	…	…	…	0-5

Fig. 19. Logical frames in TC-MAC for N=61 and k=6

For communication between two vehicles, the vehicles will use their time slots on the SCHs to exchange messages. Let's assume we have two vehicles, A with local ID 4 and B with local ID 15. If vehicle A and vehicle B are on different slot numbers on the SCHs and want to exchange messages, they can use their on time slots own the SCHs to complete the process, Figure 20.

For communication with $RSUs$, if the RSU is communicating with one vehicle, the RSU will be treated as if it is a vehicle. If the RSU is trying to communicate with more than one vehicle, the RSU and the other vehicles will use their time slots to communicate. The communications of the $RSUs$ are considered as any other vehicle in the cluster.



	← --- 1 TDMA FRAME (100 msec)---→																
	S0	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**S8**	**S9**	**S10**	**…**	**…**	**…**	**…**	**S60**	**S61**	**S62**
SCH 5	5	11	17	23	29	35	41	47	53	59	…	…	…	…	…	…	…	…
SCH 4	4 (A)	10	16	22	28	34	40	46	52	58	…	…	…	…	…	…	…	…
SCH 3	3	9	15(B)	21	27	33	39	45	51	57	…	…	…	…	…	…	…	…
SCH 2	2	8	14	20	26	32	38	44	50	56	…	…	…	…	…	…	…	…
SCH 1	1	7	13	19	25	31	37	43	49	55	61	…	…	…	…	…	…	…
SCH 0	0	6	12	18	24	30	36	42	48	54	60	…	…	…	…	…	…	…
CCH	6-11	12-17	18-23	24-29	30-35	36-41	42-47	48-53	54-59	60-65	…	…	…	…	…	…	…	0-5

Fig. 20. Example shows the slots that vehicle A with local ID 4 and vehicle B with local ID 15 can use to communicate on the SCHs

3.2 INTRA-CLUSTER COMMUNICATION

For intra-cluster communication, we look at single-hop and multi-hop clusters. Our goal is to design lightweight communication protocols that avoid, to the largest extent possible, the involvement of the CH in setting up connections between vehicles. As a single-hop cluster, all vehicles in the cluster can communicate directly; while vehicles in the multi-hop cluster need to rely on other vehicle(s) in the cluster to communicate with all vehicles. Consequently, the vehicles do not need to discover their neighbors.

Each vehicle uses its own mini-slot to disseminate status information. The first byte of the mini-slot can be used to encode $2^8 = 128$ different situations; a few of them are listed below:

- **0** indicates that the vehicle is not communicating on its own slot on the *SCH* at the moment.
- **1** indicates that the vehicle is involved in communicating with some other vehicle in the cluster on the *SCH*; the binary encoding of the ID of the interlocutor follows in the second byte.
- **2** indicates that the vehicle is involved in communicating with a multicast group in the cluster; the binary encodings of the IDs of the members of the multicast group follow in the next bytes.
- **3** indicates that the vehicle is involved in communicating with a vehicle or *RSU* outside the cluster.
- **4** indicates that the *CH* is leaving the cluster and a new *CH* is picked by the current *CH*; the binary encoding of the old ID of the new *CH* follows in the second bytes.
- **5** indicates that the vehicle is leaving the cluster.
- **6** indicates that the *CH* election process need to be performed.
- **7** indicates that the vehicle wants to join the cluster, “Handshake”. This will be sent by the new comer vehicle to the any cluster member in the targeted cluster.

- *8* is the confirmation of the "handshake" message that sent by the new comer.
- **9** indicates that the vehicle will transmit during its upcoming slots; the binary encodings of the number of frames that the vehicle will be using on its own slot on the *SCH*.
- **10** indicates that the vehicle will use its upcoming slot to transmit.

Certain messages need to be transmitted inside the cluster. These messages are safety, governance, and non-safety messages. Also, the messages could be broadcasted or unicasted. We explain our scheme below.

3.2.1 DISSEMINATING INTRA-CLUSTER SAFETY/GOVERNANCE MESSAGES

The *CH* is responsible for disseminating safety and governance messages. When a safety message needs to be broadcast to the cluster, the *CH* will use its mini-slot on the *CCH* to broadcast the message. Also, the *CH* will repeat the same safety message in any available mini-slot on the *CCH* of the same TDMA frame. The reason for repeating the same safety messages is to achieve the effect of broadcasting to the entire cluster. The *CH* may decide to disseminate safety messages to a subset of the vehicles, in which case it will also broadcast an N-bit vector, indicating which vehicles are targeted by the message; if all bits are set, the message is a cluster-wide broadcast.

In addition to safety messages, the previously-described mechanism is employed for cluster governance messages including:

- The updated value of N and multicast group setup requests.
- Channels and slot times during which the *CH* has "office hours" and will listen to individual requests.

In the case of multi-hop cluster, the *CH* will disseminate safety/governance messages to nearby cluster members and will pick a vehicle to be a relay node to other cluster members that are not in the range of the *CH*. In Figure 21, we have a three-hop cluster with a transmission range of 300 m. When the *CH* wants to disseminate safety/governance messages, vehicles up to 300 m behind and 300 m ahead of the *CH* will receive the messages directly. However, the vehicles that are located more

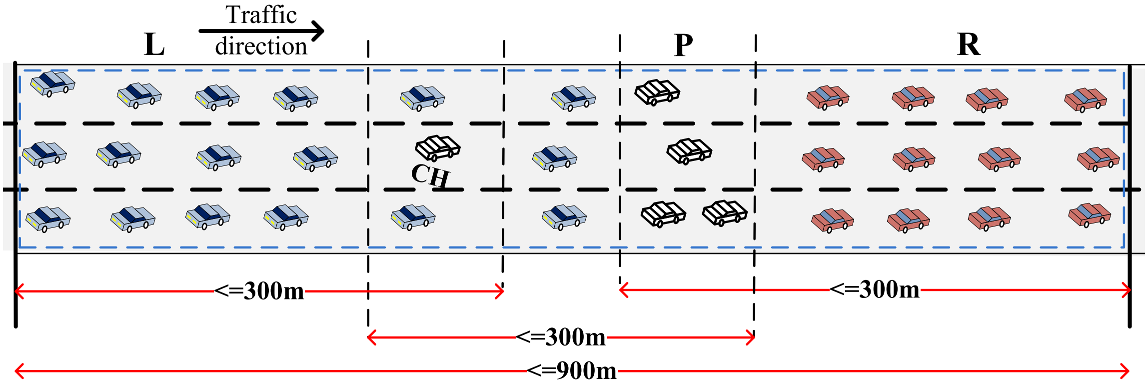


Fig. 21. Disseminating of safety messages by the *CH* in a multi-hop cluster. The *CH* sends the safety messages to vehicles in group *L* and group *P*. The *CH* will pick a vehicle from group *P* to be a relay node to vehicles in group *R*

than 300 m away from the *CH* in group *R* will not be able to receive the messages. In this case, the *CH* will find a vehicle that is in range of the *CH* and other vehicles in *R* to be a relay node. This can be done by the *CH* requesting one of the farthest vehicles ahead to disseminate the safety message to other vehicles in range. As show in Figure 21, any vehicle in group *P* can be a relay node to vehicles in group *R*.

3.2.2 INTRA-CLUSTER UNICAST COMMUNICATION

Unicast (a.k.a. point-to-point) communications in a single-hop cluster are set up without *CH* intervention. Suppose vehicle i wishes to talk with vehicle j; setting up a connection between them is done as follows:

- By tuning in to vehicle j' s own mini-slot, vehicle i determines whether or not vehicle j is available.
- If so, vehicle i transmits a handshake packet on channel $j \ mod \ k$ during time slot $\lfloor \frac{j}{k} \rfloor$.

Assuming no collision (i.e. some other vehicle may also want to talk to j), j will pick up the handshake packet and will negotiate with vehicle i the parameters of the data exchange by replying on channel $i \ mod \ k$ during time slot $\lfloor \frac{i}{k} \rfloor$; again, assuming no collision, the connection between vehicles i and j has been set up. Now, both vehicles set up the first byte of their mini-slots to indicate the status change. Once the connection has been set up, the two vehicles can communicate either in i's slot,

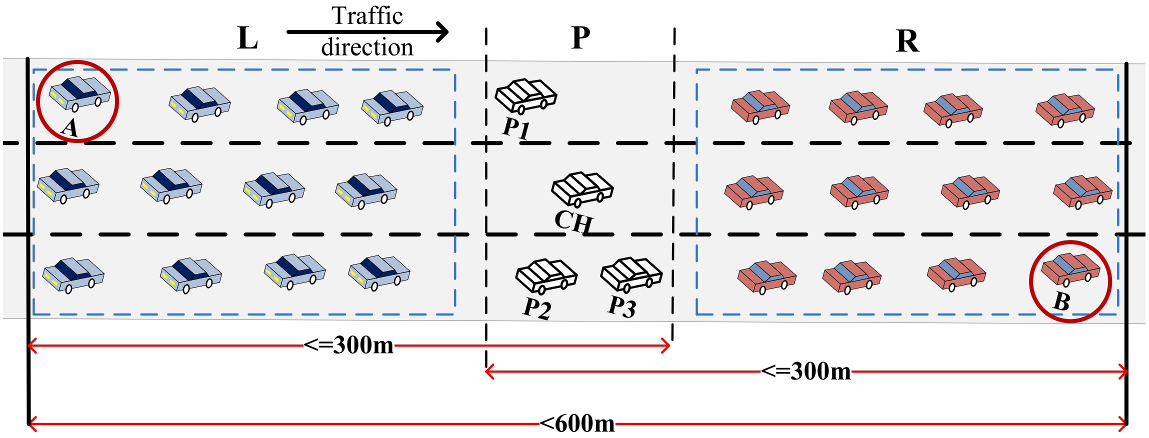


Fig. 22. Unicast of non-safety messages in a two-hop cluster, where vehicle A from group L is trying to send a non-safety message to vehicle B from group R. Vehicle A needs to find a vehicle in group P to be a relay node to vehicle B

j's slot, or both, if needed. If vehicles i and j need more than the basic amount of bandwidth, they may seek permission from the CH to use one or more extra unused time slots.

In the case of multi-hop clusters, unicast communications are set-up through negotiation with some cluster members to find a path. In this section, we will describe two multi-hop cases. The first case is a two-hop cluster. From Figure 22, we have a two-hop cluster where each vehicle in group L can communicate directly with other vehicles in group L, as well as vehicles in group P. Also, vehicles in group R can communicate directly with other vehicles in group R and group P. If vehicle A from group L wants to send a non-safety message to vehicle B from group R, vehicle A will try to find a vehicle from group P to be a proxy node for the communication between A and B. Finding vehicle in P is done by vehicle A sending a request on the CCH during A's mini-slots seeking a vehicle in the range of B. If there is a vehicle in P willing to be a relay node between A and B, vehicle P will reply to A during A's time slot on the SCH. Once the P vehicle is determined, the path is defined and vehicles can start transmission. The transmission will be done during the time slots for the three vehicles in the path, A, P and B. Figure 23 shows the process of setting up an unicast communications in a two-hop cluster.

Our goal is to complete the transmission from A to B in one TDMA frame. To achieve this goal, the use of the time slots of the vehicles needs to be in a certain order. Since it is only a two-hop cluster, we need only two TDMA time slots to transmit from *A* to *B*. However, the path from *A* to *B* has three vehicles, which means we have three different TDMA time slots. This can be done as follows:

- *A* will use the earliest time slot to send to *P*. It could be *A's*, *P's* or *B's*.
- *P* will use the earliest time slot of the remaining two time slots to send to *B*.
- If all time slots are at the same time but on different SCHs, *A* will use its own to send to *P* and then *P* will use one of the unused time slots to send to *B*.

In the case of a three-hop cluster unicast communications, the path from the sender vehicle to the receiver vehicle may involve up to four vehicles. From Figure 24, when vehicle *A* wants to send a message to vehicle *B*, two other vehicles need to be involved to complete the transmission. This can be done as follows:

- Vehicle *A* needs to find a vehicle in *P1* that is willing to participate in this transmission. This process is done in the same fashion as finding *P* in the two-hop cluster above.
- After finding a vehicle in *P1*, this vehicle looks for a vehicle in *P2* that is willing to participate in this transmission. Also, this is done as if it is a unicast of two-hop cluster.
- Once we have two vehicles from *P1* and *P2*, *A* will be able to communicate with *B*.

When the path is defined, all vehicles in the path know the TDMA time slots of each other using the local IDs of the vehicles in the path.

To achieve the transmission from vehicle *A* to vehicle *B* in one TDMA frame, the order of using the TDMA time slots for the four vehicles needs to certain way. Since we have four TDMA time slots and we need only three of them to complete the transmission, the selection of the TDMA time slots can be done as follow:

- *A* will use the earliest time slot to send to *P1*, it could be *A's*, *P1's*, *P2's* or *B's*.
- *P1* will use the earliest time slot of the remaining three time slots to send *P2*.
- *P2* will use the earliest time slot of the remaining two time slots to send *B*.

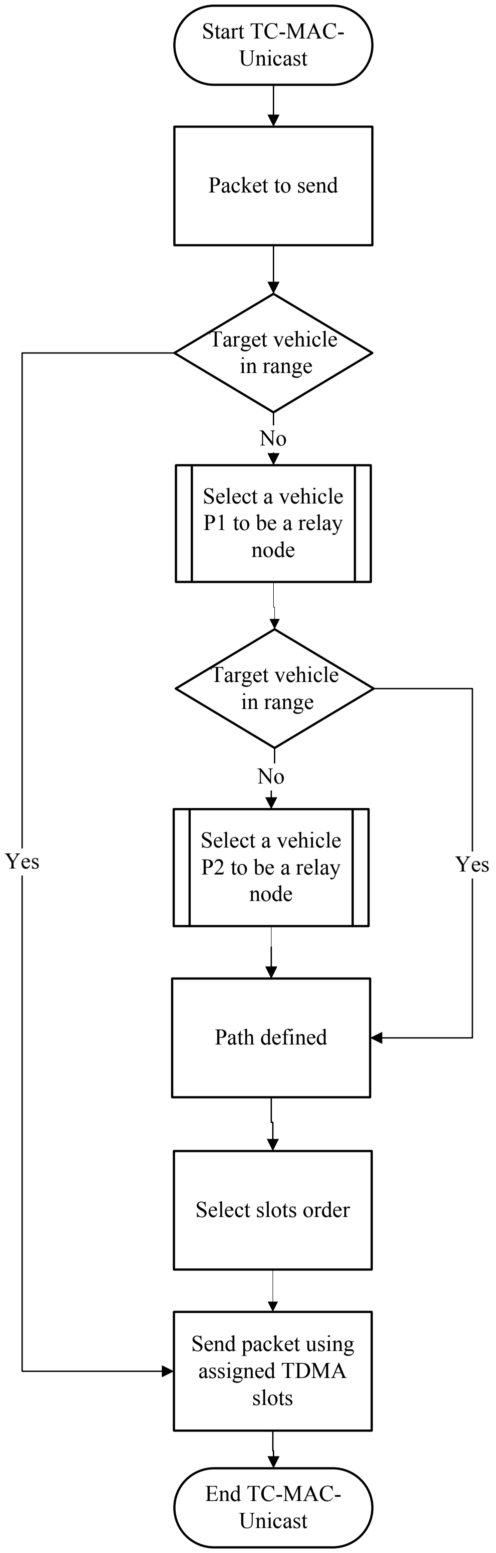


Fig. 23. Flowchart of the multi-hop intra-cluster unicast communications

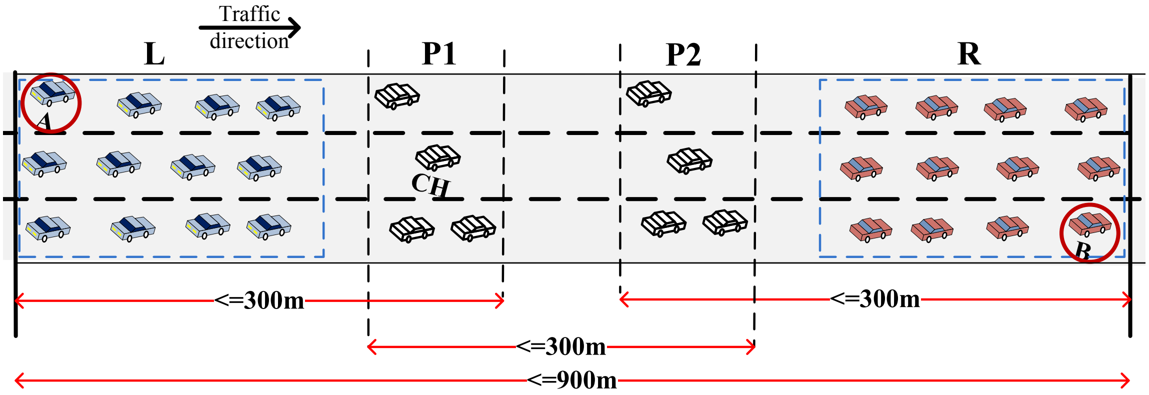


Fig. 24. Unicast of non-safety messages in a three-hop cluster

- If all time slots are at the same time but on different SCHs, unused slots will be used. If the unused slots can not make the transmission in one TDMA frame, the *CH* needs to get involved and change one of the local IDs of the vehicle in the transmission path to make transmission complete in one TDMA frame.

3.2.3 INTRA-CLUSTER MULTICAST COMMUNICATION

Multicast (a.k.a. point-to-multipoint) communications may be set up with or without *CH* intervention. Suppose vehicle j wishes to establish a multicast group involving vehicles $i_1, i_2, ..., i_p$. If the multicast group is small, vehicle j will attempt to send a handshake message to each of the remaining vehicles in the multicast group. Once the group has been set up, vehicle j will transmit on channel $j \ mod \ k$ during time slot time $\lfloor \frac{j}{k} \rfloor$ and all the other vehicles will listen to the channel. If the size of the multicast group is large, vehicle j will send the *CH* a multicast group request consisting of its own ID along with an N-bit vector with the bits corresponding to the multicast group set. Once received by the CH, this multicast group set-up request will be disseminated by the *CH* in the next available logical frame, by all the modalities discussed above. Once the multicast group has been set up, vehicle j will transmit to the group on channel $j \ mod \ k$ during slot $\lfloor \frac{j}{k} \rfloor$. For a multi-hop cluster, if the vehicles are not in the range of vehicle j, vehicle j will find a vehicle in P1 or P1 and P2 to act as a proxy(s) to the other vehicles in multicast group.

3.3 IMPACT OF GUARD INTERVAL

A *GI* is used at the beginning of each channel interval in WAVE to enable the radio devices to complete switching and account for any synchronization inaccuracy. The channel switching is measured to be 40-80 *microseconds* [44], while the GPS timing inaccuracy is in *nanoseconds* [2]. So, using *GIs* should have an impact on either the number of slots or the slot size in a TC-MAC frame. If we need to keep the number of slots in TC-MAC frame as it was before adding the *GI*, the slot size will be smaller. To calculate the new slot size, I consider only the switching time, since the GPS timing inaccuracy is in *nanoseconds*. Also, the *GIs* will be added to each slot on the *SCHs*. On the *CCH*, the *GI* will be added at the beginning of the slot, and will serve for all six mini-slots. If the switching time is 80 *microseconds*, the new *SCH* slot size will be as follows:

$$1.6msec - 80\mu sec = 1.520msec$$

The size of non-safety message using *GI* will be:

$$1.520msec = \frac{1000msec}{6,000,000b} \times Xmsec$$

$$X = 91,200b = 1,140B$$

For the mini-slot on *CCH*, the size will be as follow:

$$\frac{\text{The slot size on SCHs}}{\text{Number of mini-slots in the slot}}$$

$$\frac{1.520msec}{6}$$

$$0.253msec$$

The size of safety/update message using *GI* will be:

$$0.253msec = \frac{1000msec}{6,000,000b} \times Xmsec$$

$$X = 1,520b = 190B$$

3.4 SUMMARY

In this chapter, I presented a cluster-based TDMA scheduling protocol for MAC for VANETs (TC-MAC), in which the collision-free intra-cluster communications were organized by the *CH* using a TDMA scheme. We also explained a light weight slot reservation algorithm. Our work is based on guaranteeing that vehicles receive non-safety messages without affecting safety messages. We also changed the concept of having two intervals by having vehicles listening to the control channel and the service channels during the same time cycle. This scheme should be easy and fast to maintain. The evaluation of TC-MAC will be covered in Chapter 6.

CHAPTER 4

CLUSTERING AND CLUSTER MAINTENANCE

In this chapter, I will present two different protocols. First, I will describe in details the clustering formation protocol. This protocol is based on electing the clusterhead *(CH)* among the vehicles that present the majority of the traffic flow. Second, I will describe the cluster maintenance protocol using TC-MAC. This chapter aims to describe the design architecture as well as to provide the details of each architectural component.

4.1 CLUSTERHEAD ELECTION AND CLUSTER FORMATION

In this section, I will present a lane-based clustering algorithm designed to provide stability in cluster lifetime for VANETs. Stable clustering methods reduce the overhead of re-clustering and lead to an efficient hierarchical network topology. During the creation of VANET clusters, cluster members select one member to be the *CH*. Fewer *CH* changes result in a more stable cluster. To achieve this goal, cluster members must select a member that has the potential to be a *CH* longer than other cluster members. My method aims to select a *CH* based on the lane where most of the traffic will flow.

The proposed protocol is based on the assumption that each vehicle knows its exact lane on the road via a lane detection system and an in-depth digital street map that includes lane information, such as NAVTEQs NAVSTREETS [45]. A lane detection system is an important element of many applications in VANETs, such the Extended Emergency Brake Light system [46].

The Global Positioning System (GPS) is the primary system that is used for vehicle localization. However, GPS has weaknesses when it comes to updating the positioning data and when there is no signal. GPS has a 5 m error which is larger than the distance between lanes. There has been much research on detecting and localizing lanes on roads. Several algorithms have been proposed using different techniques. Some methods use GPS combined with a wheel odometer [47], which provides relative localization as it detects changes in pose relative to the previous

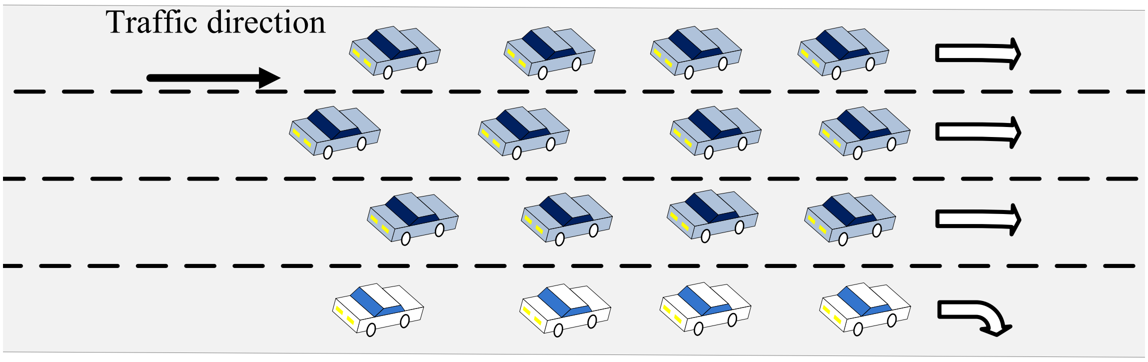


Fig. 25. Showing how the traffic flow is dividing vehicles ahead in different directions. The rightmost lane present an exit on the highway

pose. The advantage of the wheel odometer is that it is high resolution and simple to use. It can typically detect movements on the order of tenths of millimeters. Other algorithms do not use GPS and instead use techniques such as vision [48], [49], [50], LIDAR (Light Detection and Ranging) [51], and a beacon network using infrastructure to triangulate vehicle position [52].

In my approach, I considered highway scenarios with exits in the design of the proposed protocol. The *CH* will be selected based on the flow of the majority of traffic. For example, if the highway has four lanes and three of them are going straight and one of them is for the exit on the highway, the *CH* should be selected from the lanes that are going straight. This research applies the knowledge of each vehicle's lane and the flow direction of each lane (Figure 25).

In highway scenarios, traffic flow may split at each exit. There are three main traffic flows in a highway: Left Exit (*LE*), Right Exit (*RE*), and No Exit (*NE*). The highway may have all three types of traffic flows or only some of them at a certain point on the highway. *LE* is applied to the leftmost lane(s) if it splits the traffic to the left, *RE* is applied to the rightmost lane(s) if it splits the traffic to right, while *NE* is applied to the lane(s) where the traffic goes straight.

This proposal follows the same general idea as the Utility Function [28], but employs a different set of rules. We considered the effect of traffic flow, using lane information, on the process of *CH* election. Each vehicle computes and broadcasts its *CH* Level (*CHL*) along with its update message on the *CCH*. The vehicle with the highest *CHL* will be elected as the *CH*. If the elected *CH* is a member of another cluster, the vehicles will choose the second highest, etc. *CHL* is defined as

$$CHL_i = NCL(t)_i + ADL_i + AVL_i \tag{1}$$

where *NCL* is the network connectivity level, *ADL* is average distance level, and *AVL* is average velocity level. The computation of each of these metrics is described below.

4.1.1 LANE WEIGHT

The key to our approach is to consider the lane a vehicle belongs to. We apply to each metric a lane weight (*LW*) for each traffic flow (*LE, RE* and *NE*). The weight is determined based on the total number of lanes on the highway (*TNL*) and the number of lanes for each traffic flow (*NLTF*). If the road has three different traffic flows, we will have three different *LWs*. *LW* is defined as

$$LW_k = \frac{1}{TNL} \times NLTF_k \tag{2}$$

where k is the lane number. For example, if we have a road of four lanes where one lane is *LE*, one lane is *RE*, and two lanes are *NE*, then the *LW* for each traffic flow will be $LW_{LE} = LW_{RE} = 0.25$ and $LW_{NE} = 0.50$. If a vehicle is on a lane with traffic flow *LE*, then it will use LW_{LE}. In the equations that follow, LW_{TF} represents the *LW* for the traffic flow of the vehicle performing the computation.

4.1.2 NETWORK CONNECTIVITY LEVEL

To compute the Network Connectivity Level *(NCL)*, we need to calculate the overall *NCL* and the *NCL* for each traffic flow. The overall *NCL*, α, is the maximum number of vehicles that are directly connected to vehicle i. This is defined as

$$\alpha_i(t) = \sum_i A(i, j, t) \tag{3}$$

where j is a potential neighbouring vehicle. *A(I, j, t)* is equal to 1 if a connection between i and j exists at time t, and is equal to 0 otherwise. At this point, we have calculated the connectivity level between a vehicle and all other vehicles on the road. Now, we calculate the connectivity level for a vehicle and the vehicles in the traffic flow it belongs to. The traffic flow connectivity level β for vehicle i is defined as

$$\beta_i(t) = \sum_{J_{TF}} A(i, J_{TF}, t) \tag{4}$$

where J_{TF} is a vehicle in the same traffic flow as vehicle i.

After calculating both levels of network connectivity, we define the total connectivity level for vehicle i on a lane belonging to traffic flow TF as

$$NCL_i(t) = \beta_i(t) + \alpha_i(t) \times LW_{TF} \tag{5}$$

where LW_{TF} is the lane weight for the lane that vehicle i occupies.

4.1.3 AVERAGE DISTANCE LEVEL

To calculate the Average Distance Level *(ADL)*, we calculate the overall average absolute distance, δ, between vehicles that are directly connected to vehicle i. This is defined as

$$\delta_i = \frac{\sum_j \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}}{NV} \tag{6}$$

where j is any vehicle connected to i, and NV is the total number of vehicles that are directly connected to i in any lane.

Next, we calculate the average absolute distance, χ_i, between vehicle i and other vehicles in the same traffic flow, *TF*. This is defined as

$$\chi_i = \frac{\sum_{j_{TF}} \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}}{NV_{TF}} \tag{7}$$

where j is any vehicle in the same traffic flow and connected directly to i, and NV_{TF} is the total number of vehicles that are directly connected to i and in the same traffic flow.

The ADL for vehicle i in traffic flow TF is defined as

$$ADL_i = (\chi_i + \delta_i)^{-1} \times LW_{TF} \tag{8}$$

4.1.4 AVERAGE VELOCITY LEVEL

We calculated the overall Average Velocity Level (AVL) as the difference between the average velocities of all vehicles in range and the candidate CH velocity. Then, we add this to the product of LW and the average velocity for the traffic flow. The overall AVL, σ_i, for vehicle i is defined as

$$\sigma_i = \sum_j |Vel_i - Vel_j| \tag{9}$$

where j is a potential neighbouring vehicle, and Vel_i is the velocity of vehicle i.

Now, we calculate the *AVL* for vehicle i and the traffic flow it belongs to. This is defined as

$$\rho_i = \sum_{j_{TF}} |Vel_i - Vel_j| \tag{10}$$

where j_{TF} is a vehicle in the same traffic flow as vehicle i.

The *AVL* for vehicle i in traffic flow *TF* is defined as

$$AVL_i = (\rho_i + \sigma_i)^{-1} \times LW_{TF} \tag{11}$$

After calculating the values above, the *CHL* is determined for each vehicles individually, and will be broadcasted in the update message of each vehicle. Every vehicles is a candidate to be a *CH*, but the vehicles with highest *CHL* and not a member of another cluster will be elected as a *CH*. When the *CH* is elected, it will pick the cluster local ID 1. Other vehicles in the cluster will be assigned to local IDs by the *CH*. Once the *CH* has assigned all vehicles with local IDs, it will broadcast a table of the new assignment to all vehicles in the cluster on the *CCH*. This table will contain all the cluster members and their local IDs. Also, this table will be broadcasted by the *CH* every time the cluster topology changes.

If there are only two vehicles in the highway, the *CH* will be selected as the vehicle of the lowest ID. If more vehicles are joining the cluster, my *CH* election algorithm will be performed.

4.2 CLUSTER MAINTENANCE

Due to the movement of vehicles, the cluster will not stay the same for long time. The behavior of many vehicles may change the topology of the cluster; for example:

- A clusterhead leaving the cluster
- A new vehicle joining the cluster
- A cluster member leaving the cluster

- Two clusters get in range of each other
- A multi-hop cluster shrinks to a one-hop cluster

All these changes in the cluster topology will be addressed.

4.2.1 CLUSTERHEAD LEAVING THE CLUSTER

When a *CH* is elected using our *CH* election algorithm, the *CH* will be locally assigned to ID 1, as the local ID TC-MAC protocol. If the *CH* predict changes in its mobility behavior that might lead to being an unstable *CH*, it will prepare for giving up its responsibility as a *CH*. This process can be done by choosing another cluster member that is willing to serve as a stable *CH* for the cluster. From Figure 26, we have two-hop cluster where the *CH* is assigned to a local ID of 1. In this case, the *CH* is ready to resign. Before this happens, the *CH* will pick one of the vehicles in area C to be a *CH*. The reason for that is to have a two-hop cluster even after the current *CH* leaves. The process will be done by switching the local ID between the current *CH*, ID 1, and the new *CH*. If there are no vehicles in C besides the *CH*, a new three-hop cluster will be formed.

From Figure 26, C includes *CH* and vehicles x and y. Vehicle with ID z is in between x and y but not part of C. We want to prove that z is part of C and can be a *CH*. Let us assume the following:

- V is a set of vehicles in an area of 600m or less (assuming the communication range is 300m).
- *CH* is a vehicle that can communicate directly with all vehicles in V.
- C is set of vehicles that can do the job of *CH*.
- x is the farthest vehicle of C behind *CH*.
- y is the farthest vehicle of C ahead of *CH*.
- L is a set of vehicles that are behind *CH* and not part of C.
- R is a set of vehicles that are ahead of the *CH* and not part of C.

Given:

$$\forall c \in C \text{ Adjust to all other vehicles in } V$$

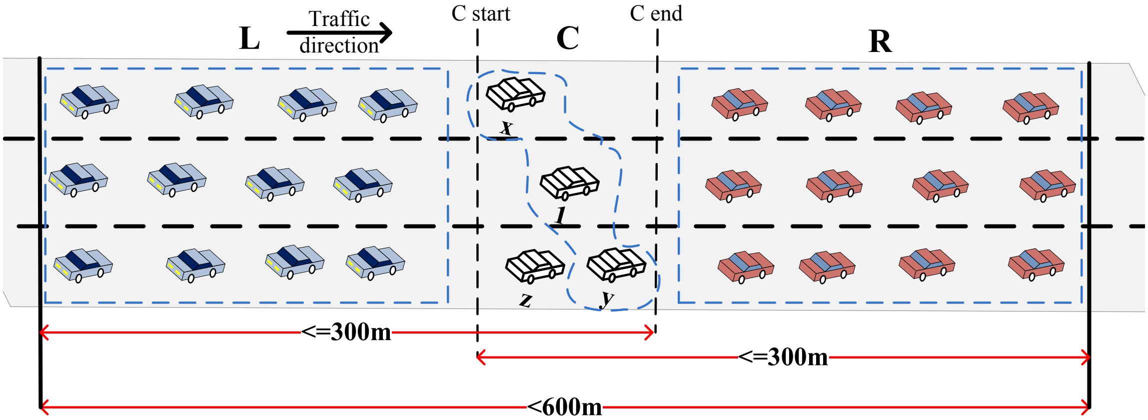


Fig. 26. *CH*, ID 1, is about to resign. So, it will choose vehicle x, y, or z to be a new *CH*

R.T.P:

Any vehicle z between x and y is part of C and can be a *CH*.

Given:

$$1. \, (z - l) \leqslant 300m$$

where $l \in L$

$$2. \, (z - r) \leqslant 300m$$

where $r \in R$

Proof 1:

$$\because x \in C \tag{12}$$

$$\because x \in \; [C_{start}, \; C_{end}] \tag{13}$$

where C_{start} is the start point of C and C_{end} is the end point of C.

$$\because |C - R| \leq 300m \tag{14}$$

Assume we have vehicle r that belongs to R where:

$$|C - r| \leq 300m$$

From 12, we can replace C with x:

$$\therefore |x - r| \leq 300m \tag{15}$$

$$\because |x - C_{end}| \geq |z - C_{end}| \tag{16}$$

$$\therefore |z - r| \leq 300m \tag{17}$$

Proof 2:

$$\because y \in C \tag{18}$$

$$\because y \in [C_{start}, C_{end}] \tag{19}$$

where C_{start} is the start point of C and C_{end} is the end point of C.

$$|C - l| \leq 300m$$

From 18, we can replace C with y:

$$\therefore |y - l| \leq 300m \tag{20}$$

$$\because |y - C_{start}| \geq |z - C_{start}| \tag{21}$$

$$\therefore |z - l| \leq 300m \tag{22}$$

From 17 and 22, z can be a *CH*.

If there are no vehicles in C besides the *CH*, two single-hop clusters will be formed. This process is done by selecting a vehicle from L and a vehicle from R to act as *CHs* for the new clusters, cluster L and cluster R. The selection of these two vehicles is done by the current *CH* informing vehicles in L and R that they will be two single-hop clusters. And then, the *CH* election algorithm will be performed in both new clusters, L and R, by cluster members sending their own Clusterhead Level *CHL* value using their own mini-slot from the old cluster on the *CCH*. The old *CH* will try to join one of the new formed cluster.

4.2.2 A NEW VEHICLE JOINING THE CLUSTER

Now, I will describe the process of accepting a new vehicle in the cluster. Importantly, this operation is preempted by the broadcast of update messages. I describe this process for a single-hop cluster and two-hop cluster.

For the single-hop cluster, when a new vehicle in the same direction on the highway wishes to join the cluster, it will attempt to get the *CH's* attention by transmitting in the mini-slot of virtual vehicle 0. Assuming that it was successful in getting the *CH's* attention, the *CH* will broadcast a “New Vehicle” message as discussed above and tentatively assign the newcomer ID 0. The new-comer will then transmit a "Handshake" message in both the *CH's* slot of the current frame as well as the virtual vehicle's slot in the next frame; this has the effect of broadcasting to the entire cluster. To confirm that the single-hop structure of the cluster is preserved, each vehicle in the cluster will have to confirm receipt of the "Handshake". To achieve this, each vehicle sets the first byte of its mini-slot to 8 and the *CH* will read all the mini-slots. If all the vehicles have confirmed receipt of the "Handshake" message, the new vehicle is accepted in the cluster and will be allocated the lowest available ID number. If necessary, N will be adjusted and the *CH* will broadcast the updated information to the other vehicles in the cluster. If the cluster size has reached the maximum number of vehicles, the new vehicle will be informed that it cannot join the cluster and it will be treated as if it is from another cluster. Figure 27 shows when vehicle i is trying to join the cluster.

For the two-hop cluster, assuming that the new vehicle is not in range of the *CH* but in range of other vehicles in the cluster, when a new vehicle in the same direction on the highway wishes to join the cluster, it will attempt to get the attention of any vehicle in the cluster by transmitting in the mini-slot of virtual vehicle 0. Assuming that it was successful in getting the attention of one of the cluster members, the cluster member will inform the *CH* of the newcomer using the cluster member's mini-slot. When the *CH* receives a newcomer notification from a cluster member, the *CH* will start to look for an available local ID to assign to the newcomer and then send it back to the cluster member that discovered the newcomer. Once the cluster member receives the available ID from the *CH*, it will inform the newcomer in the same way as in the single-hop cluster. All the communications between the cluster member and the *CH* for assigning the local ID to the newcomer are done using their own mini-slots on the *CCH*. Figure 28 shows when vehicle i is trying to join the cluster by communicating with cluster member j on the mini-slot 0.

4.2.3 A CLUSTER MEMBER LEAVING THE CLUSTER

When a vehicle i predicts mobility changes that might lead it to leave the cluster, it will broadcast that to the entire cluster using its own mini-slot. If vehicle i fails to do so, the *CH* will assume that vehicle i is gone after a certain period of time of not hearing from vehicle i. Once the *CH* defines that vehicle i is no longer a cluster member, the *CH* will place the local ID of i in the list of the available IDs. If a new vehicle wants to join the cluster, the *CH* can assign the new vehicle to one of the

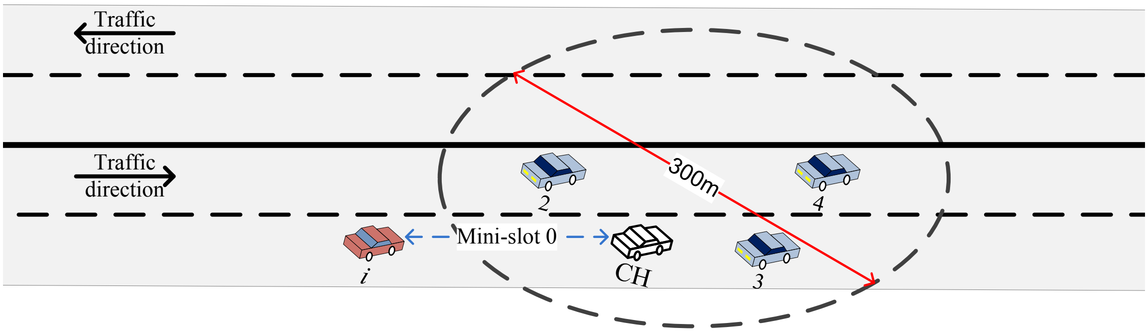


Fig. 27. A new vehicle i is trying to join a single-hop cluster by communicating with the cluster's CH using mini-slot 0. Notice that vehicle i is traveling on the same direction as the cluster

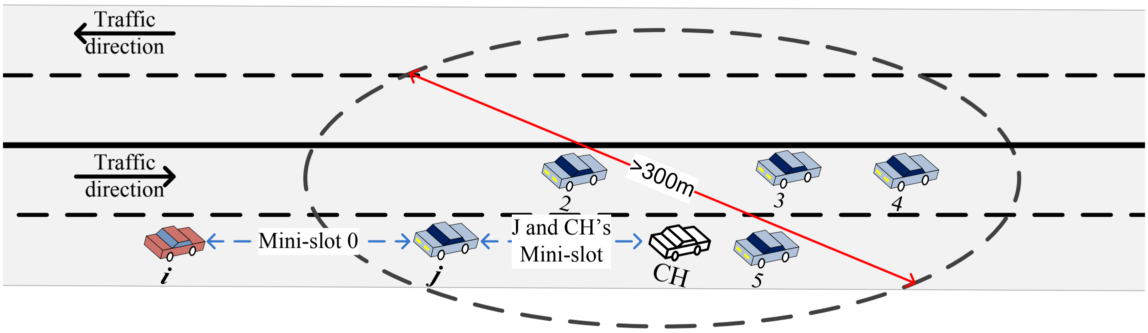


Fig. 28. A new vehicle i is trying to join a two-hop cluster by communicating with the cluster member j using mini-slot 0. Notice that vehicle i is traveling on the same direction as the cluster

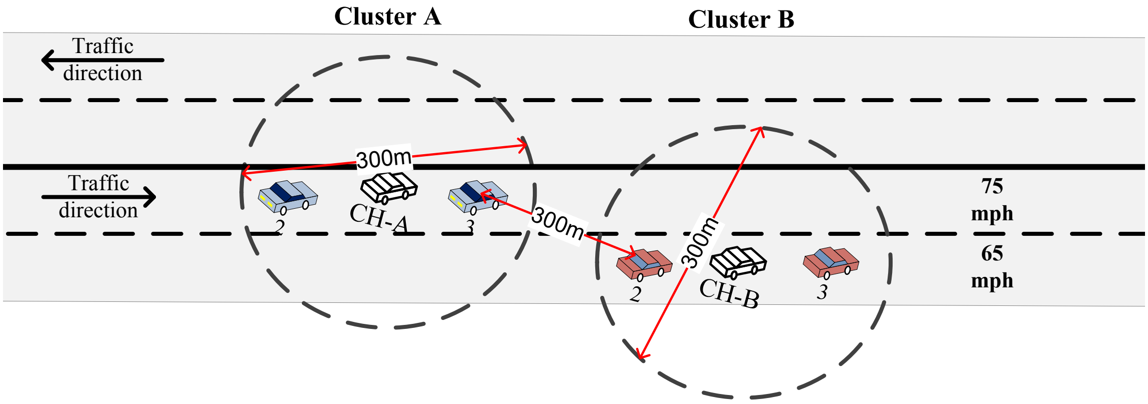


Fig. 29. Two single hop clusters in range of each other, before merging

available local IDs. Also, the *CH* will send an update table of the cluster members and their local IDs.

4.2.4 MERGING TWO CLUSTERS

In my scheme, the cluster is considered to be a single-hop or two-hop. So, because of the dynamics of vehicles on the road, two single-hop clusters may come into range of each other to form a two-hop cluster.

When two single-hop clusters are in range of each other and the vehicles are traveling in the same direction, these two clusters can merge creating one single-hop or multi-hop cluster, as shown in Figure 29. The cluster members that are located at the head and tail of each cluster will be the vehicles that get involved in the merging process. When these vehicles detect the present of the other cluster, they will start the merging process. The merging process will be performed by vehicles advertising themselves to other vehicles in the different clusters by sending an advertisement messages that includes their *IDs* and their clusters size during the mini-slot of the virtual vehicle, ID 0. If the total number of vehicles in both clusters is less than or equal to the maximum number of the cluster size in TC-MAC, the merging process will continue. This procedure detecting outsiders will be performed periodically, e.g. 1 second. Once the messages are exchanged between the two vehicles of the two different clusters, each one of them will try to communicate with the other based on each one's mini-slot of the original cluster. During this communication, both vehicles will inform their *CHs* about the merging process.

The vehicle in the larger cluster will be the *CH*. Then, the old *CH* in the large

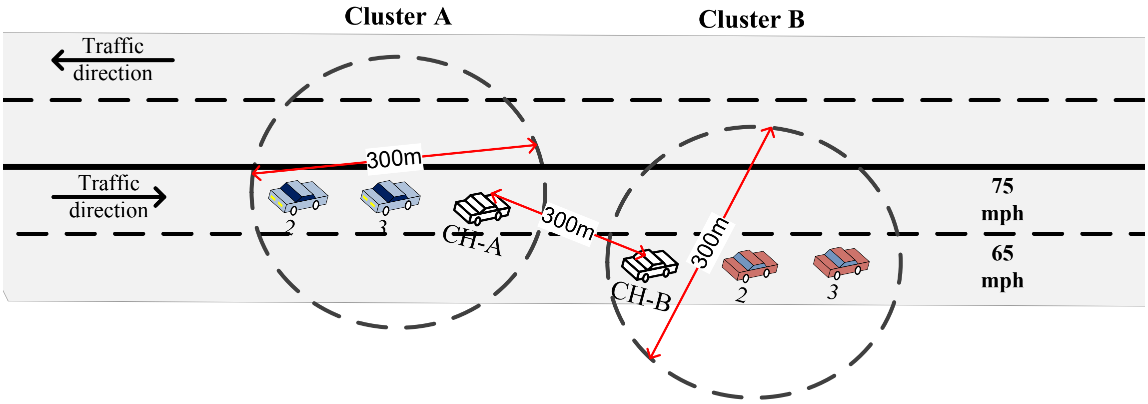


Fig. 30. Closest vehicles from each cluster swapping IDs with their *CHs* during the process of merging

cluster will resign by swapping IDs, ID 1, with the new *CH*. The old *CH* will be pushed into a stack for future needs, as shown in Figure 30. The sender vehicle in the other cluster will swap its ID with the *CH* of its own cluster, and the old *CH* will be pushed into a stack for future needs. The reason for pushing the old *CHs* in a stack is that they will be valuable if the two single-hop clusters split again. Once the whole large cluster is formed, the vehicles of the old large single-hop cluster will have the same IDs, while the vehicles of the old small single-hop cluster will receive new IDs. The new IDs will be the old IDs plus the size of the old larger one-hop cluster. So, the low IDs will be in the old large one-hop cluster and the high IDs will be in old the smaller one-hop cluster, as shown in Figure 31. If the clusters are of the same size, the *CH* of the approaching cluster will be the *CH* of the new cluster. If the new large cluster seemed to be stable for a long time, not splitting into the two old clusters, every vehicle will calculate the *CHL* value and send it along with its update message to elect a new *CH*.

4.2.5 A MULTI-HOP CLUSTER SHRINKING TO A ONE-HOP CLUSTER

When vehicles in a multi-hop cluster move closer to each other, there is a chance of forming a single-hop cluster (Figure 31). In this case, vehicles will keep the same IDs as the *CH* remains a *CH*. Also, vehicles in the cluster will be able to communicate directly with other vehicles in the cluster, as shown in Figure 32.

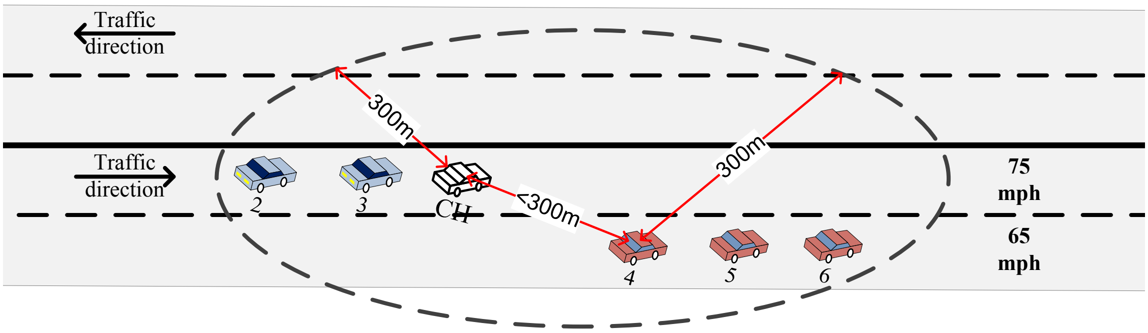


Fig. 31. The new *CH* for the new cluster that resulted of the merge

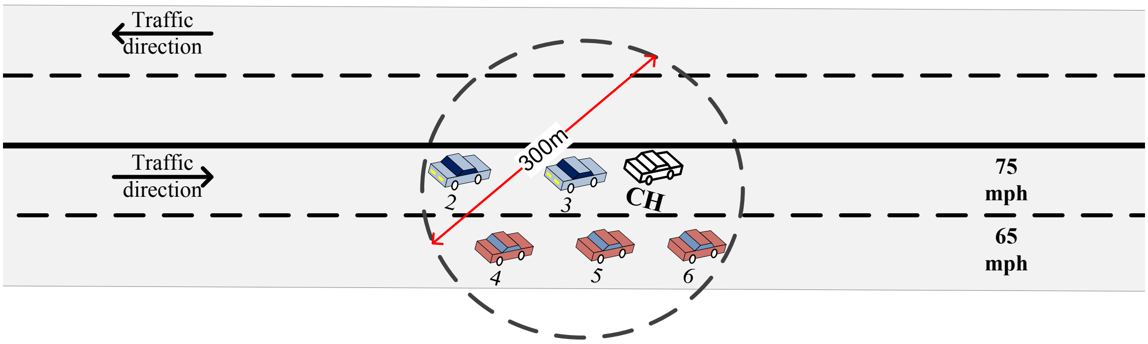


Fig. 32. One single-hop cluster, after shrinking of a multi-hop cluster

4.3 SUMMARY

I presented an algorithm for *CH* election based on the traffic flow of vehicles on the highway. With the availability of lane detection, lane direction and map matching, we were able to select the most stable clusterhead. I also presented a cluster maintenance algorithm. With the availability of local IDs, we were able to have less overhead when the topology of the cluster changes. We tested our algorithm using simulation. An evaluation of these technique will be presented in Chapter 6.

CHAPTER 5

CDMA/TC-MAC HYBRID PROTOCOL FOR INTER-CLUSTER COMMUNICATIONS

In this chapter I present a CDMA/TC-MAC hybrid protocol for inter-cluster communications. In this protocol, a Code Division Multiple access (CDMA) scheme is implemented on top of TC-MAC to enable vehicles to communicate with vehicles in different clusters and with RSUs. In addition, the hidden and exposed terminal problems are addressed by the proposed protocol.

5.1 CDMA/TC-MAC ARCHITECTURE

As explained in Chapter 3, the TC-MAC protocol is able to manage the transmission of safety/update and non-safety messages inside one cluster. However, VANETs require broadcasting of safety/update messages to nearby vehicles from different clusters. Besides that, VANETs need to be able to transmit non-safety messages to any other vehicles on the road, if needed. TC-MAC, as it is, is not able to communicate with other vehicles from different clusters. To solve this issue, TC-MAC needs to be modified to overcome the inter-cluster communications challenge. The modification I made is using CDMA combined with TC-MAC for inter-cluster communications, resulting in CDMA/TC-MAC. This addition does not have any impact on the performance of TC-MAC, in terms of intra-cluster communication.

The CDMA protocol type used in CDMA/TC-MAC is a transmitter-based protocol. In this type of CDMA protocol, a transmission code is assigned to each cluster to be used for intra-cluster communications. With the use of the CDMA assigned code and TC-MAC inside the cluster, intra-cluster collisions should not happen. Also, this will support broadcast inside the cluster without the risk of the interference with other vehicles from other clusters.

In order for CDMA/TC-MAC to work, it needs two different protocols. These protocols are the Code Assignment protocol and the Recovery protocol. I will explain these two protocols in the following subsections.

5.1.1 CODE ASSIGNMENT PROTOCOL

The Code Assignment protocol is used to assign a CDMA code to the cluster. This task is done by the *CH*. There are 8 different CDMA codes in CDMA/TC-MAC. These codes are used based on the highway and the cluster's direction. CDMA/TC-MAC considers vehicles in opposite directions, as well as on opposite highways. With the information provided by GPS, vehicles can determine their direction; if a vehicle is moving from west to east (or north to south), it is in the right direction (R) and opposite vehicles are in the left direction (L). For the direction of the highway, CDMA/TC-MAC uses the same method as the direction of the vehicles, but instead of using GPS information, it uses a digital map that tells the highway direction. For example, highway I-95 goes from north to south, and it will always be considered as from north to south even if the highway is curving. Figure 33 shows the directions of the vehicles and the highways, similar to what has been used in VeMAC [5] but with addition of highways. The reason for considering the directions of the highways is to avoid data collisions when two highways intersect with each other.

Table 5 shows the different codes that are used in CDMA/TC-MAC. During the cluster formation and the *CH* election process, vehicles on the right direction highway (R) will use CDMA/TC-MAC code 1 to exchange messages that contain their Clusterhead Level *CHL* value. Code 1 is assigned to be a global code for vehicles on right direction highways. It has several uses, which I explain as needed. Once the *CH* is elected, the *CH* will select a CDMA/TC-MAC code and send it to the cluster members using code 1. If the cluster direction is R, the *CH* will choose a CDMA/TC-MAC code from 2, 3, and 4, otherwise the *CH* will choose a code from 5, 6, and 7. The code selected by the *CH* should be different than the codes of the ahead and behind clusters. If the newly formed cluster can not find out about the codes of the other clusters, the *CH* will pick any code from the codes in its direction. After determining the CDMA/TC-MAC code for the cluster, the cluster member will use TC-MAC for intra-cluster communications, as explained in Chapter 3.

In order to minimize problems caused by the variability of the cluster range, CDMA/TC-MAC assures the existence of two clusters with two different codes between clusters sharing the same code. In this case, three CDMA/TC-MAC codes are necessary for each direction on the same highway. From Figure 34, if the range of cluster B, and the gap between vehicle X in cluster A and vehicle Y in cluster C is less than 300 m, a collision will happen between vehicles X and Y if they were

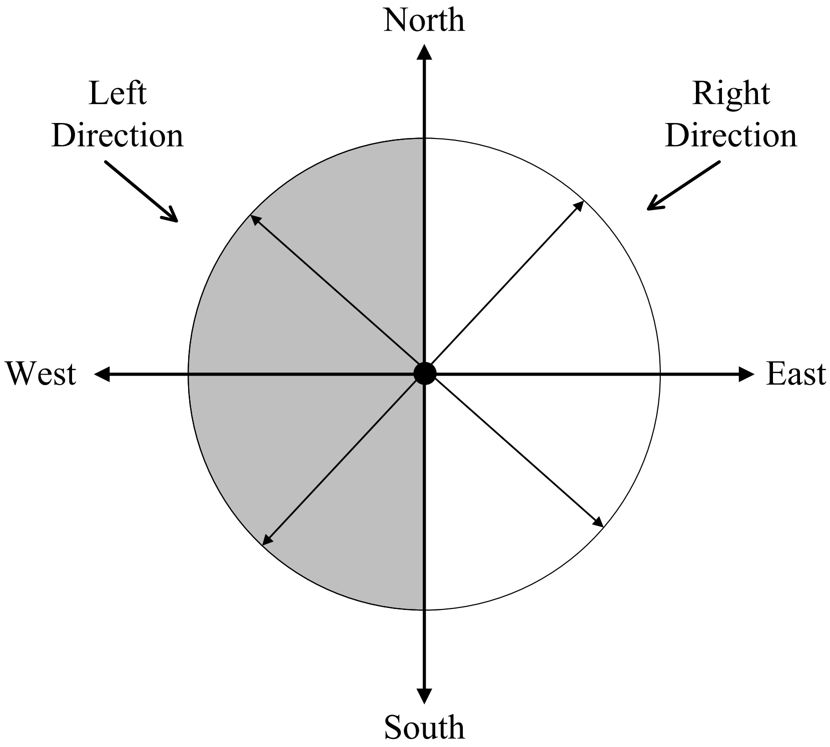


Fig. 33. Vehicles and highway directions in CDMA/TC-MAC. Vehicles and highways in the dark area are Left direction, while others are Right direction

using the same code. This collision will not happen between vehicle X and vehicle Z because they have two clusters in between, which results in having a gap that is larger than 300 m.

5.1.2 RECOVERY PROTOCOL

The Recovery Protocol in CDMA/TC-MAC is designed to solve the issue of having two clusters in range of each other and sharing the same code, which will lead to interference between vehicles in both clusters. This issue will not happen unless both clusters are on the same highway and travelling in the same direction.

The Recovery Protocol works when a vehicle in a cluster detects another vehicle from another cluster using the same CDMA code. From Figure 35, there are two clusters, cluster A and cluster B. When the gap between the two clusters is less than 300 m, at least one vehicle from each cluster will detect the present of the other cluster, vehicles X and Y. If both clusters have the same CDMA code, the vehicle X from the approaching cluster will inform its CH about the situation, cluster A.

TABLE 5. CDMA/TC-MAC codes

CDMA/TC-MAC Code	Highway Direction	Vehicle Direction
1	R	R and L
2, 3, and 4	R and L	R
5, 6, and 7	R and L	L
8	L	R and L

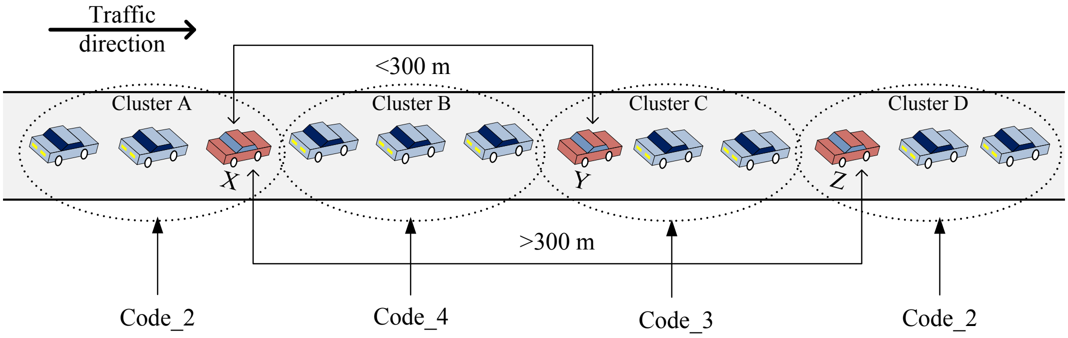


Fig. 34. Three CDMA codes are needed for each direction in CDMA/TC-MAC to avoid having two different clusters in range of each other and using the same CDMA code

Then the *CH* of cluster *A* will pick a different CDMA code than the old one and broadcast it to the members of its cluster. The newly picked CDMA code also needs to be different than the code of the cluster behind, if there is one.

If there is an intersection of two highways, and two clusters from both highways sharing the same CDMA code are passing the intersection, interference between vehicles will happen. The recovery protocol addresses this issue. Let us assume that there are two clusters, cluster *A* on a *R* highway and cluster *B* on a *L* highway, and both clusters are using CDMA code 2. When they get to the intersection point, each cluster will detect the other cluster. The cluster on highway *R* will change its CDMA code, while the cluster on highway *L* will keep its CDMA code. The changing of code is done in the same way as if we have two adjacent clusters sharing the same CDMA code.

5.2 DISSEMINATING INTER-CLUSTER MESSAGES

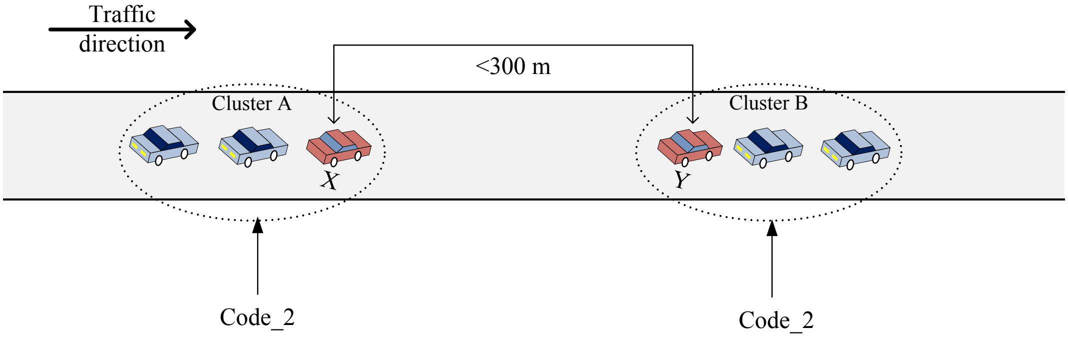


Fig. 35. Two clusters are using the same CDMA code, Code 2. When the gap between both cluster is less than 300 m, a code conflict will be detected and the recovery protocol will start

After explaining the architecture of CDMA/TC-MAC, now I will explain how to disseminate messages between two adjacent clusters. The goal of CDMA/TC-MAC is to enable the transmission of all types of messages between clusters, with avoiding the network issues. In the following, I will explain the setup process needed for the inter-cluster communications in CDMA/TC-MAC.

5.2.1 SETUP

For all types of messages, two adjacent clusters can exchange messages without the risk of collisions. To do so, every cluster on the road has to do the following:

- The *CH* needs to select at least two vehicles to monitor for other clusters.
- The selected vehicles must be located at the head and the tail of the cluster to act as gateways (GWs) of the cluster.
- The GWs will switch to the *CCH* during the mini-slot 0 (virtual vehicle mini-slot in TC-MAC) using the global code of the highway (1 or 8) to send/receive advertisement to/from other GWs of other clusters.
- During the switching to mini-slot 0, the GWs that are located at the head of the clusters will be listening for advertisements from the other GWs that are located at the tail of the clusters. During the following frame, the GWs that are located at the head of the clusters will be sending their advertisements to the other GWs that are located at the tail of the clusters. This should solve

the issue of having collision when two *GWs*, head and tail, are trying to send advertisements at the same time.

- The advertisement of the *GWs* includes the following information:
    - The CDMA code of the *GW's* cluster.
    - The cluster size.
- Each *GW* will inform its own *CH* about the discovery, assuming both clusters are using 2 different CDMA codes.
- The *CHs* will assign a local ID as in TC-MAC to the *GWs* of the opposite cluster to act as a member of both clusters, assuming no clusters merge. The new local IDs will be picked from the upper bound IDs of the cluster, which should help the *GWs* not miss safety/update messages from cluster members of their original clusters.

Once the process is completed, the *GWs* will switch between the CDMA codes of the two clusters during the TDMA time slot they are assigned.

For more explanation, Figure 36 shows two adjacent clusters, *A* and *B*, within less than 300 *m* gap and with two different CDMA codes. In order for cluster *A* and cluster *B* to communicate, the *GWs* of both clusters need to get engaged in the process. Let us assume that GW_{tail} of cluster *A* is sending and GW_{head} of cluster *B* is receiving at time *t* on the mini-slot of *CCH* and using the CDMA global code of *Code_1*. At time *t*:

- GW_{tail} of cluster *A* will send an advertisement on the mini-slot 0 on the *CCH* using CDMA code *Code_1*. This advertisement will include the cluster size of *A* and the CDMA code used by cluster *A* (3, *Code_2*).
- GW_{head} of cluster *B* will receive the advertisement from GW_{tail} of cluster *A*. Then, it will pass the advertisement to the *CH* of *B* during its own time slot in cluster *B* using CDMA code *Code_3*. Note, this process will be done within 100 *msec*, before GW_{head} of cluster *B* sends back its own advertisement-acknowledgement to GW_{tail} of cluster *A* on the *CCH*.

At time *t+100 msec*:

- GW_{head} of cluster B will send an advertisement-acknowledgement to GW_{tail} of cluster A. Beside its cluster's CDMA code and size, GW_{head} of cluster B indicates that it has received the previous advertisement from GW_{tail} of cluster A, and is in the process of assigning a local ID in cluster B to GW_{tail} of cluster A.

- The GW_{tail} of cluster A will receive the advertisement-acknowledgement from GW_{head} of cluster B, and then pass its own *CH*. For the next transmission cycle, GW_{tail} of cluster A should listen to the GW_{head} of cluster B on *CCH* during mini-slot 0 to get its own local ID in cluster B.

At time *t+200 msec*:

- By this time, GW_{head} of cluster B has already received from the *CH* of cluster B the local ID assigned to GW_{tail} of cluster A.

- GW_{head} of cluster B sends the local ID to GW_{tail} of cluster A.

- By the end of this cycle, GW_{tail} of cluster A should receive the local ID of GW_{head} of cluster B in cluster A, and it can be sent to GW_{head} of cluster B during its own time slot in cluster B.

After finishing the setup for the inter-cluster communications, vehicles GW_{tail} of cluster A and GW_{head} of cluster B will be treated as cluster members of both clusters A and B.

For the *RSUs*, the setup to join a cluster is done in very similar way to the setup of inter-cluster communications. The difference is that the *RSUs* will be always listening for an advertisement from *GWs* of clusters on the highway. Before joining any cluster, the *RSU* will be using the global CDMA code of the highway. Once it detects an advertisement from a *GW*, it will send back an advertisement-acknowledgement asking for a local ID in the cluster. When the local ID is assigned to the *RSU*, the RSU will be treated as a cluster member.

5.2.2 SAFETY AND NON-SAFETY MESSAGE DISSEMINATION

For the dissemination of safety messages, if a dangerous situation is detected in one cluster, other clusters can be inform of the situation. This can be done as follows, assuming the setup for inter-cluster communications is completed between clusters:

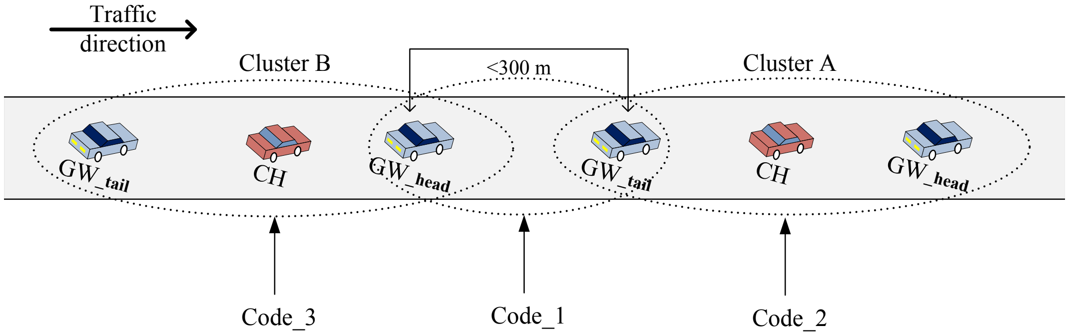


Fig. 36. Two adjacent clusters with less than 300 m gap are wishing to communicate. The process will be completed by exchanging several setup messages between GW_{tail} from cluster A and GW_{head} from cluster B

- The detected dangerous situation will be broadcasted to the cluster members where it happened, as explained in Chapter 3, including the GWs.
- If the GW is connected to or in the path of the intended cluster, the GW will rebroadcast the safety message in the other cluster during its own mini-slot in the other cluster and using the CDMA code of the other cluster.

For non-safety message dissemination, once the inter-cluster communication setup is completed, the process should be clear. If a cluster member needs to send a non-safety message to another vehicle in a different cluster, it will do the following:

- The vehicle that is wishing to send a non-safety message will communicate with the GW of its cluster that is in the path of the destination vehicle.
- If the GW is available on its SCH time slot, it will participate in the delivery of the message. Otherwise, the sender needs to wait or find another GW.

5.3 SUMMARY

In this chapter, I presented a CDMA/TC-MAC hybrid protocol for inter-cluster communications in VANETs. This protocol allows two neighboring clusters to communicate without effecting the intra-cluster communications in each cluster. It uses different CDMA codes for each cluster that are different than the CDMA code of the neighboring cluster. There are 8 different codes, 3 codes for each direction traveling north to south, and 3 codes for each direction traveling east to west. For each highway direction, there is one global code, codes 1 and 8, that can be used by the clusters to setup the inter-cluster communications. This protocol also explains the process of communicating with the RSUs.

CHAPTER 6

EVALUATION

This chapter describes the evaluation of the proposed protocols in this dissertation. The evaluation is done using network simulations under different traffic scenarios. I evaluated the clusterhead election, cluster formation, and TC-MAC protocols. I will explain each one of them in a different section in this chapter.

6.1 METHODOLOGY

This section describes the experimental setup and methodology used to evaluate the performance of the clusterhead election and cluster formation protocol, and the TC-MAC protocol, including the network and traffic scenarios used in the evaluation. The evaluation metrics will be described later for each protocol. I assumed all vehicle on the road are equipped with GPS and DSRC transceivers.

6.1.1 NETWORK CONFIGURATION

I ran simulations using ns-3 network simulator [53], which is a follow-on to the popular ns-2 simulator. For VANET, I used modules [54] that added well-known traffic mobility models, the Intelligent Driver Model (IDM) [55] and the MOBIL lane change model [56]. The goal was to create a vehicular network on highways with different number of lanes and different number of vehicles.

Clusterhead Election and Cluster Formation

The network parameters used in the simulation of the clusterhead election and cluster formation are listed in Table 6. I set the transmission range for vehicles to 300 m [57]. The transmission rate is set to be 6 Mbps, which was shown to be the optimal data rate for VANETs [57]. The update messages are of size 200 bytes, which is a reasonable size for safety/update messages [57]. These messages include the basic vehicle's information, such as speed, position, direction, velocity, and lane on the road. Besides the basic information, the vehicle will include its Clusterhead Level *(CHL)*. The *CHL* is calculated by every vehicle trying to join the cluster based

TABLE 6. Network parameters used in the simulation for the clusterhead election and cluster formation protocol

Parameter	Values
TX Range	300 m
Update Message size	200 bytes
Data Rate	6 Mbps

on the calculations explained in Chapter 4. When the vehicles are the process of creating the cluster, they will be sending their update messages using CSMA/CA mechanism to access the medium.

TC-MAC

The network parameters used to evaluate TC-MAC as compared to WAVE are listed in Table 7. I set the transmission range for vehicles to 300 *m*, which a common transmission range that is used in VANETs [57]. So in the case of a single-hop cluster, the maximum length of the cluster will be 300 *m*, and it will be 600 *m* for the two-hop cluster. For safety/update messages, the maximum message size is 200 bytes [57]. Since the slot size on the SCHs in TC-MAC is 6 times the size of the mini-slot on the CCH, the maximum non-safety message size will be 1,200 bytes. I set the data rate to 6 Mbps, which is shown to be the optimal data rate for VANETs [57]. For WAVE, I set the CCHI and SCHI to be 50 *msec* each, which will make the transmission cycle to be 100 *msec* [11].

In TC-MAC, all vehicles in the cluster are using their own time slots to communicate with other vehicles. These slots are assigned to them by the *CH*, to communicate with other vehicles; this assignment mechanism is explained in Chapter 3. For WAVE, all vehicles are using CSMA/CA mechanism to access the medium.

6.1.2 TRAFFIC SCENARIOS

In this part of this chapter, I will explain the scenarios I used in the simulations. The scenarios are divided into two parts. First part, explains the scenarios used for the clusterhead election and cluster formation protocol; while the second part explains the scenarios used for TC-MAC.

TABLE 7. Network parameters used in the simulation

Parameter	Values for WAVE	Values for TC-MAC
TX Range	300 m	300 m
Max. Safety Message size	200 bytes	200 bytes
Max. Non-Safety Message size	1200 bytes	1200 bytes
Data Rate	6 Mbps	6 Mbps
CCHI	0.05 sec	N/A
SCHI	0.05 sec	N/A

Clusterhead Election and Cluster Formation

The scenarios implemented for the highway are with different number of lanes and the same traffic density. I evaluated scenarios with 2, 3, and 4 lanes. For the traffic density, I used 50 vehicles. The highway length is 5,000 m. I used two exits in the highway. The exits are right exits and located at the 1,500m and 3,000m marks. The speed limit is set to 29 m/sec.

Since all the exits are right exits, only vehicles on the rightmost lane are able to take the exit. I assumed 25% of vehicles on the rightmost lane will take the first exit, and 25% of vehicles on the rightmost lane will take the second exit.

TC-MAC

The scenarios implemented for the highway are with different number of lanes and different density levels. I evaluated scenarios with 2, 3, and 4 lanes. For the traffic density levels, I used four different levels; they are Low, Med, High, and Very High. Table 8 shows the number of vehicles per lane for each density level, as well as the gap between vehicles in the lane and the speed limit. As the gap between vehicles increases, the number of vehicles in the lane decreases, and this will affect the density level on the highway. The highway length is set to 5,000 m and vehicles are set to different speed limits. The speed limits are set to maintain particular density of the vehicles in the road.

The simulation for the clusters is done by generating vehicles that are moving at a speed that maintains the stability of the cluster. In other words, if the density level is low, which means the gap between vehicles is large, the speed limit of vehicles is

TABLE 8. Density levels in the highway with the speed limits

Density Level	Number of Vehicles per Lane	Gap between Vehicles	Speed Limit
Low	5	60 m	29 m/sec
Med	12	20 m	10 m/sec
High	20	10 m	5 m/sec
Very High	50	1 m	0 m/sec

set to high. Table 8 shows the different speed limits used in the simulations based on the traffic density. The relationship between the speed of vehicles and the traffic density is inverse. For example, in very high traffic density, the gap between vehicles is 1 *m* and the speed limit is 0 m/sec. In the very high density case, the vehicles are generated and the first vehicles in every lane make a complete stop at 1000 *m* mark.

Several scenarios have been tested for TC-MAC and WAVE using single-hop clusters and two-hop clusters. A single-hop cluster is a cluster where all cluster members can talk to each other directly, while a two-hop cluster is a cluster where some cluster members need a relay node to communicate with other cluster members. There are 12 scenarios that are implemented for TC-MAC and WAVE using a single-hop cluster as shown in Table 9. For the two-hop cluster, I implemented another 12 different scenarios for both TC-MAC and WAVE, as shown in Table 10. Since I am limited with the maximum number of vehicles in the cluster, based on the network settings, I used 372 vehicles in the cluster instead of 400 vehicles when the traffic density is very high and the number of lanes is 4. The other 28 vehicles can form their own cluster. For the two-hop cluster in WAVE, vehicles are acting as in the single-hop cluster. On other words, all vehicles will try to send messages whenever they need and the air is clear to send.

6.2 CLUSTERHEAD ELECTION AND CLUSTER FORMATION

In this section, I will present the evaluation of my clusterhead election and cluster formation protocol. The goal was to create experiments in a highway with different number of lanes. I ran experiments to measure the stability of the cluster, in terms of the number of times the clusterhead changes. I compared the performance of

TABLE 9. Scenarios for TC-MAC and WAVE using a single-hop cluster, the maximum cluster length is 300 m

Scenario	Density Level	Number of Lanes	Maximum Number of Vehicles in the Cluster	Communication Density
1	Low	2	10	100
2	Med	2	24	240
3	High	2	40	400
4	Very High	2	100	100
5	Low	3	15	150
6	Med	3	36	360
7	High	3	60	600
8	Very High	3	150	1500
9	Low	4	20	200
10	Med	4	48	480
11	High	4	80	800
12	Very High	4	200	2000

my algorithm with the Lowest-ID clustering, the Highest-Degree, and the Utility Function algorithms.

6.2.1 EVALUATION METRICS

The metric used to evaluate the clusterhead election algorithm is measuring the stability of the cluster by counting the number of *CH* changes. Before each exit, I use the election algorithm to choose the *CH* and observe if the *CH* changes for the majority of traffic after the exit.

6.2.2 EVALUATION

In this part, I will evaluate the performance of the Clusterhead election algorithm and compare it to the Lowest-ID clustering algorithm, the Highest-Degree algorithm, and the Utility Function algorithm. I will evaluate each algorithm under all scenarios with 10 different runs for each scenario.

Since my clusterhead election algorithm is based on having the knowledge of the

TABLE 10. Scenarios for TC-MAC and WAVE using a two-hop cluster, the maximum cluster length is 600 m

Scenario	Density Level	Number of Lanes	Maximum Number of Vehicles in the Cluster	Communication Density
1	Low	2	20	200
2	Med	2	48	480
3	High	2	80	800
4	Very High	2	200	2000
5	Low	3	30	300
6	Med	3	72	720
7	High	3	120	1200
8	Very High	3	300	3000
9	Low	4	40	400
10	Med	4	96	960
11	High	4	160	1600
12	Very High	4	372	3720

vehicle's lane, the Lane Weight (LW) of each lane needs to be calculated; this can be done for each lane in the highway as explained in Chapter 4. In the case of 2-lane highway, the leftmost lane is NE lane and the rightmost lane is RE lane. So, the values of LW_{NE} and LW_{RE} are equal to 0.5. For the 3-lane highway, there are 2 NE and 1 RE. The value of $LW_{NE} = 2/3$ and the value of $LW_{RE} = 1/3$. For the 4-lane highway, there are 3 NE and 1 RE. The value of $LW_{NE} = 3/4$ and the value of $LW_{RE} = 1/4$.

Figure 37 shows the number of CH changes for each clusterhead election algorithm after the first exit. It is clear that my clusterhead election algorithm (labelled Traffic Flow) generally performed better than the others. I noticed that in the case of the 2-lane highway, my algorithm and the Utility Function suffered from two CH changes. The reason for that is my algorithm treated both lanes equally. In other words, the LW for the NE and RE are the same.

Figure 38 shows the number of CH changes for each clusterhead election algorithm after the second exit. The results show that my clusterhead election algorithm performed better than the other algorithms. The CH using my algorithm did not

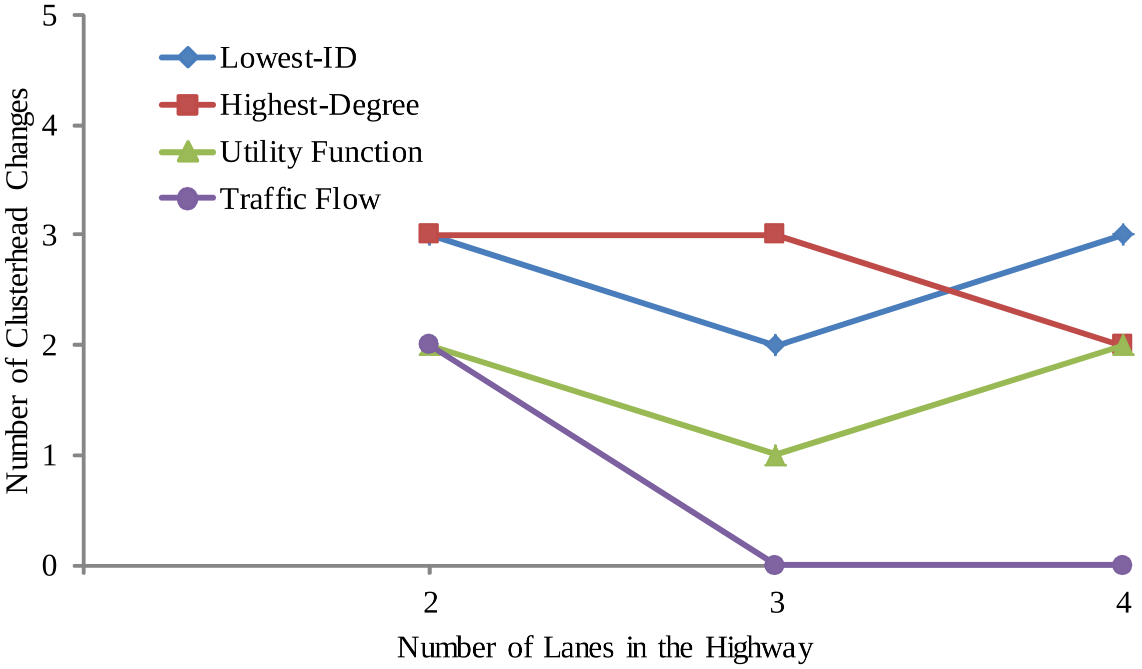


Fig. 37. Clusterhead changes vs. number of lanes in the highway after the first exit

change after the second exit in 3-lane and 4-lane highways.

The overhead of the clusterhead election process is as simple as update messages broadcasted by each vehicle in order to signal existence of itself to all its neighbors. As a result of receiving the update messages from all neighboring vehicles, each vehicle is able to dynamically build up its latest neighbor list and calculate the *CHL* and send it with the next update message.

6.2.3 SUMMARY

I presented an algorithm for clusterhead election based on the traffic flow of vehicles in the highway. With the availability of lane detection, lane direction and map matching, this algorithm was able to select the most stable clusterhead. I tested this algorithm using a highway with a two exit scenario and followed the elected clusterhead passing the two exits. This algorithm showed longer clusterhead lifetime than the Lowest-ID, Highest-Degree and the Utility Function algorithms.

6.3 TC-MAC PROTOCOL

In this section, I will present the evaluation of TC-MAC. The goal was to run

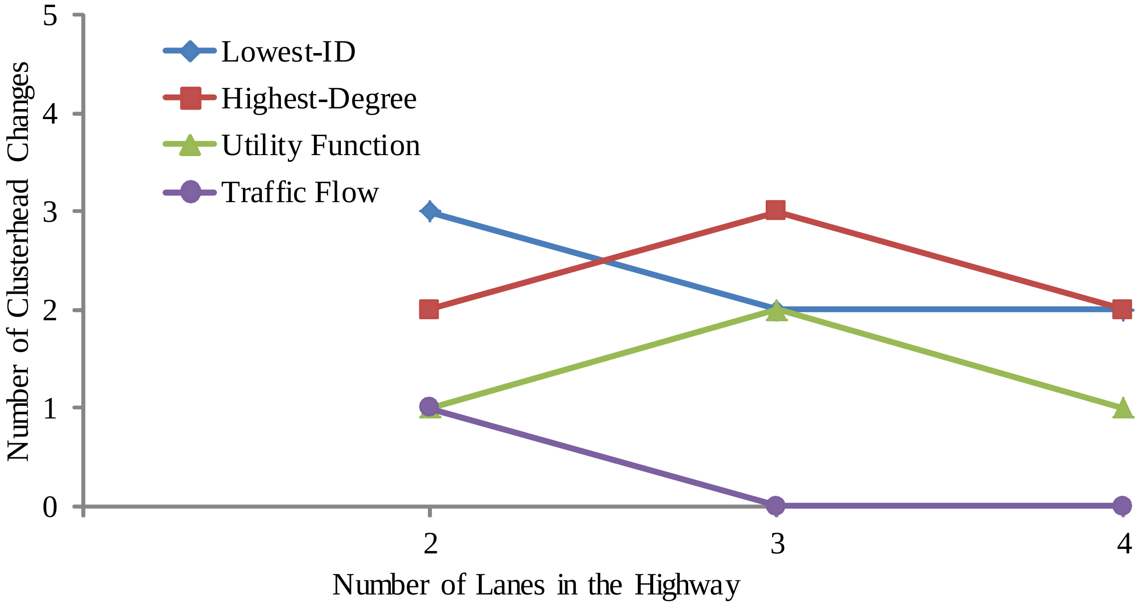


Fig. 38. Clusterhead changes vs. number of lanes in the highway after the second exit

experiments on a highway with different levels of vehicle densities and evaluate TC-MAC as compared with WAVE standard. I ran 5 runs and calculated the average for each scenario and measured the reliability of safety messages and the throughput of non-safety messages.

Safety/update messages are periodically broadcast information to surrounding vehicles. These messages provide every vehicle in the cluster with accurate and timely information about their neighbors. Update messages are very important in detecting the possibility of unsafe situations, while safety messages are important in informing the vehicles about the unsafe situations. Therefore, safety/update messages need to be sent by each vehicle in every TC-MAC frame, every 100 *msec* [11]. In the experiments of evaluating the reliability of safety/update messages, every vehicle is trying to send a safety/update message every 100 *msec*. Besides being broadcasted at a high generation rate, safety/update messages require a high reliability level. Achieving a high level of reliability is a challenge for broadcasting safety/update messages. Before running the scenarios of the single-hop cluster, I calculated the communication density (CD) [58] on the road. CD is used to measure the channel load in vehicular communications. This is done by calculating the number of carrier sensible events per unit of time. Table 9 shows the CD of the single-hop clusters, and Table 10 shows the *CD* of the two-hop clusters. In the two-hop clusters, the *CD* of the vehicles that are in the middle and in range of all the cluster members is calculated as if they all were in a single-hop cluster. For example, if we have a two-hop cluster of the size of 372, and we have a vehicle that can hear every vehicle in the cluster, the *CD* would be 3720.

6.3.1 TC-MAC FRAME

With the transmission cycle of 100 *msec*, the frame size of TC-MAC is set to be 100 *msec*. Based on the above network configuration used in the simulation, we can find out the maximum number of vehicles that can be in one cluster using TC-MAC. To calculate the maximum number of vehicles in the cluster, we need to find the time needed to transmit a 200 byte safety message on the CCH, or the time needed to transmit a 1,200 byte non-safety message on the SCHs.

For a 200 byte safety message, the time needed to be transmitted on the CCH is:

$$1600b \times \tfrac{1{,}000msec}{6{,}000{,}000b} = 0.267msec$$

And for a 1,200 byte non-safety message, the time needed to be transmitted on any of the SCHs is:

$$96,000b \times \tfrac{1{,}000msec}{6{,}000{,}000b} = 1.6msec$$

Since the maximum slot size on the SCHs is 1.6 *msec* and the frame size is 100 ms, the number of slots on each SCH in one TC-MAC frame is:

$$\lfloor\frac{FrameSize}{SCHMaximumSlotSize}\rfloor = \lfloor\frac{100msec}{1.6msec}\rfloor = 62slots$$

Since DSRC has 6 different SCHs, the number of vehicles allowed to be in one cluster in TC-MAC is 6 × 62 = 372 vehicles. Figure 39 shows the layout of the TC-MAC frame used in my simulations.

	← ---------- 1 TC-MAC FRAME (100 msec) ---------- →																	
	S0	**S1**	**S2**	**S3**	**…**	**…**	**…**	**S33**	**S34**	**…**	**…**	**…**	**…**	**…**	**…**	**S59**	**S60**	**S61**
SCH 5	5	11	17	23				203	209							359	365	371
SCH 4	4	10	16	22				202	208							358	364	370
SCH 3	3	9	15	21				201	207							357	363	369
SCH 2	2	8	14	20				200	206							356	362	368
SCH 1	1	7	13	19				199	205							355	361	367
SCH 0	0	6	12	18				198	204							354	360	366
CCH	6~ 11	12~ 17	18~ 23	24~ 29	…	…	…	204~ 209	210~ 215	…	…	…	…	…	…	360~ 365	366~ 371	0~ 5

Fig. 39. TC-MAC frame based on maximum safety message size of 200 bytes and maximum non-safety message size of 1,200 bytes.

6.3.2 TC-MAC EVALUATION METRICS

The metrics used to evaluate TC-MAC and WAVE are to measure the reliability of safety/update messages and the throughput of the non-safety messages. The metrics are as follows:

1. Reliability of safety messages: Here I measure the percentage of the successful delivery of safety/update messages for both TC-MAC and WAVE. For TC-MAC, I did the measurement in two ways, direct and indirect messages. The direct safety/update messages are the messages that are received without being rebroadcasted by the *CH*, while the indirect safety/update messages are the ones that are received after being rebroadcasted by the *CH*.

2. Throughput of non-safety messages: Here I measure the the throughput of the non-safety messages on the *SCHs*. For TC-MAC, I calculated the optimal and the the worst case throughput. The optimal throughput is when every vehicle in the cluster is sending a non-safety message to another cluster member that have different slot number on the *SCHs*. On other words, in every time slot on the *SCHs*, all six vehicles are sending non-safety messages and some other vehicles in the cluster are receiving the same messages. The worst case throughput is when we have three vehicles of the same slot number on the *SCHs* sending non-safety messages to the other three vehicles on their slot on the *SCHs*. In the simulation, when the vehicle is sending a non-safety message, the receiving vehicle is picked randomly. In some cases, the sending and the receiving vehicles were on the same time slot on the *SCHs*. For WAVE, every vehicle is trying to send a non-safety message during the *SCHI*. The safety messages are on the *CCH* during each vehicle's mini-slot in TC-MAC and during the *CCHI* in WAVE. The throughput was measured in terms of *kbps*.

   In the simulation, I use a single-hop cluster. So, the non-safety messages are going from the source vehicle to the destination directly and without the need of having another cluster member to act as a relay node. The reason is here I am trying to measure the throughput of the network, without the dealing with routing issues.

6.3.3 SINGLE-HOP CLUSTER EVALUATION

In the single-hop cluster, all cluster members are in range of each other. So, every vehicle in the cluster is able to send and receive messages directly from other vehicles in the same cluster. Based on the transmission range used in the simulations, every vehicle in the single-hop cluster is within 300 m away from any other vehicle in the same cluster.

Reliability of Safety/Update Messages

In this part of this chapter, I show the performance of TC-MAC and WAVE, in terms of safety/update message reliability. The results are displayed in two different ways, direct and indirect safety/update messages.

- Direct Safety/Update messages

  For the results of direct safety/update message delivery, I measured the percentage of missed direct messages. If the cluster size is 15 vehicles, the number of direct safety/update messages should be 14 messages, one for each vehicle per 100 *msec*. Figure 40 shows the performance of TC-MAC under different communication densities, assuming that all vehicles in the cluster are engaged in some communication during their own *SCH* time slot. When the density is low, the percentage of the missed direct messages between vehicles is high in TC-MAC. The reason for that is switching to the *SCHs*. For example, if there are 15 vehicles in a single-hop cluster, these vehicles will be assigned to local IDs from 1 to 15. So, vehicles with local IDs 1 to 5 will be using slot 0 of TC-MAC frame on *SCHs* 1 to 5, and vehicles with local IDs 6 to 11 will be using the mini slots on the *CCH* during slot 0 of the same TC-MAC frame. If vehicles 1 to 5 are using their time slots on the *SCHs*, they will miss all the safety/update messages sent by vehicles 6 to 11. This happens only when all the vehicles in the cluster are engaged in some communication during their own *SCH* time slot. TC-MAC has addressed this issue by having the *CH* resend the needed safety messages during the unused slots on the *CCH*. When at least half of the vehicles in the cluster are engaged in communication during their own *SCH* time slot, the percentage of missed direct safety/update messages decreases. Figure 41 shows the performance of TC-MAC using a single-hop cluster when only half of the vehicles in the cluster are using their slot on the *SCHs*. If the vehicles in the cluster are less involved in communications on the

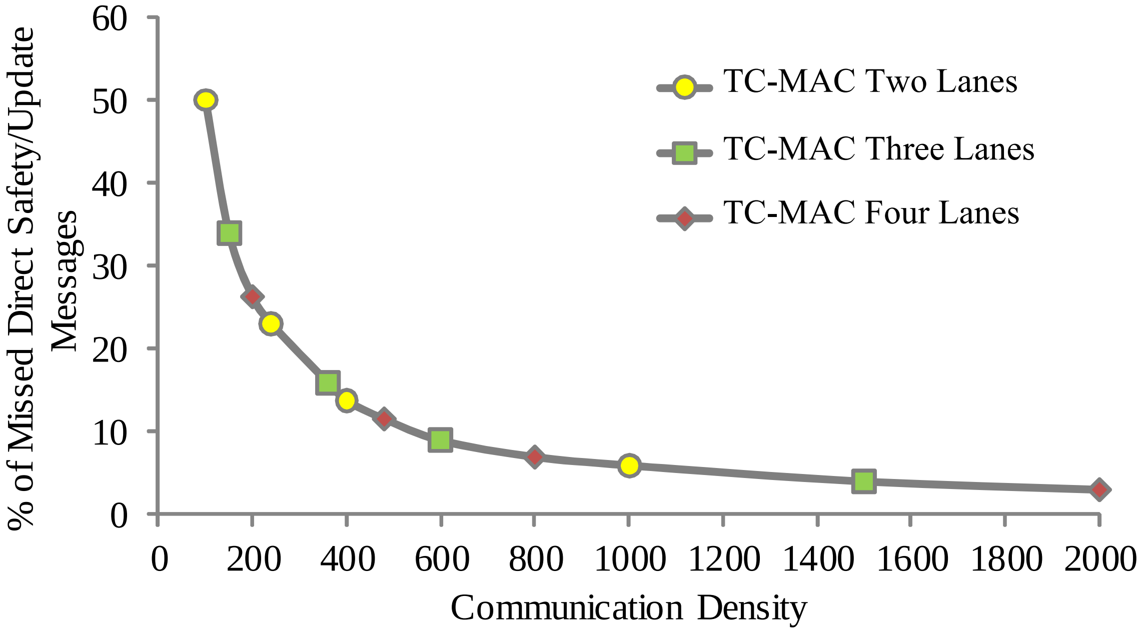


Fig. 40. Percentage of missed direct messages for TC-MAC in a single-hop cluster based on the communication density. All vehicles in the cluster are engaged in communications during their own slot time on the *SCHs*

SCHs, the percentage of the missed direct safety/update messages decreases. Unfortunately, if a vehicle is always active on its own slot on the *SCH*, this vehicle will miss all the update messages from all vehicles that have their mini-slots at the same time as the active vehicle, see Figure 39. Figure 42 shows the results when when all vehicles and half vehicles in the cluster are engaged in communications during their own slot time on the *SCHs*, it is clear that the performance is better by factor 2 when we have half of vehicles in the cluster are communicating on the *SCHs*.

On the other hand, for WAVE, I measured the percentage traffic collision on the *CCH*. Figure 43 shows the performance of WAVE when the *CCHI* is 0.05 *sec* based on the number of vehicles in terms of traffic collision. The percentage of collisions on the *CCH* increases as the traffic density increases. The reason for this is the increase of messages that need to be sent during the *CCHI*.

- Indirect Safety messages

For the indirect safety/update messages, TC-MAC rebroadcasts safety messages but not update messages. Missing an update message is not as critical as

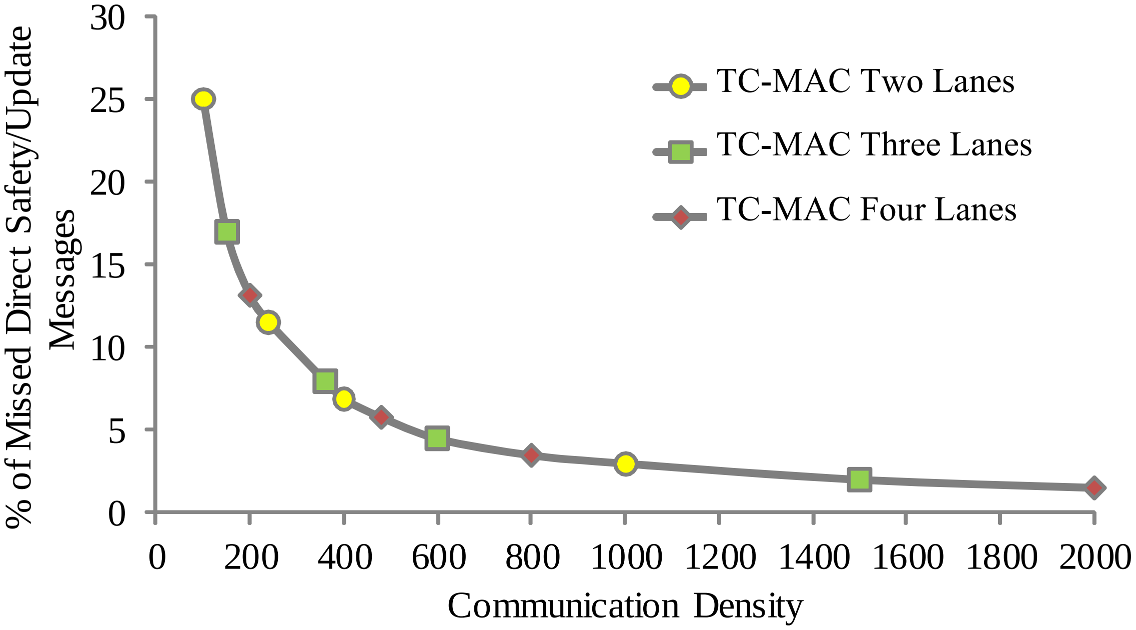


Fig. 41. Percentage of missed direct messages for TC-MAC in a single-hop cluster based on the communication density. Half of the vehicles in the cluster are engaged in communications during their own slot time on the *SCHs*

a safety message. Vehicles can predict the positions and the movement of the surrounding vehicles from the previous update messages. However, because of their importance and short lifetime, safety messages need to have high reliability. The process of rebroadcasting the safety messages in TC-MAC is done by the *CH* during its own mini-slot and all unused mini-slots on the *CCH*. In the case of the single-hop cluster, the maximum number of vehicles in the scenarios I tested was 200 vehicles. The *CH* was able to rebroadcast the safety messages to the cluster members using 173 unused mini-slots on the *CCH*, including its own. All cluster members were able to receive every missed direct message within 100 *msec*.

On the other hand, in WAVE, there is no rebroadcast by the *CH*. When the vehicle is trying to send its safety/update message, and it observes a collision on the *CCH*, the vehicle will try to send the message again during the same *CCHI*. Figure 44 shows that if the density of vehicles is high, the *CCH* gets more congested, which will reduce the reliability of safety messages.

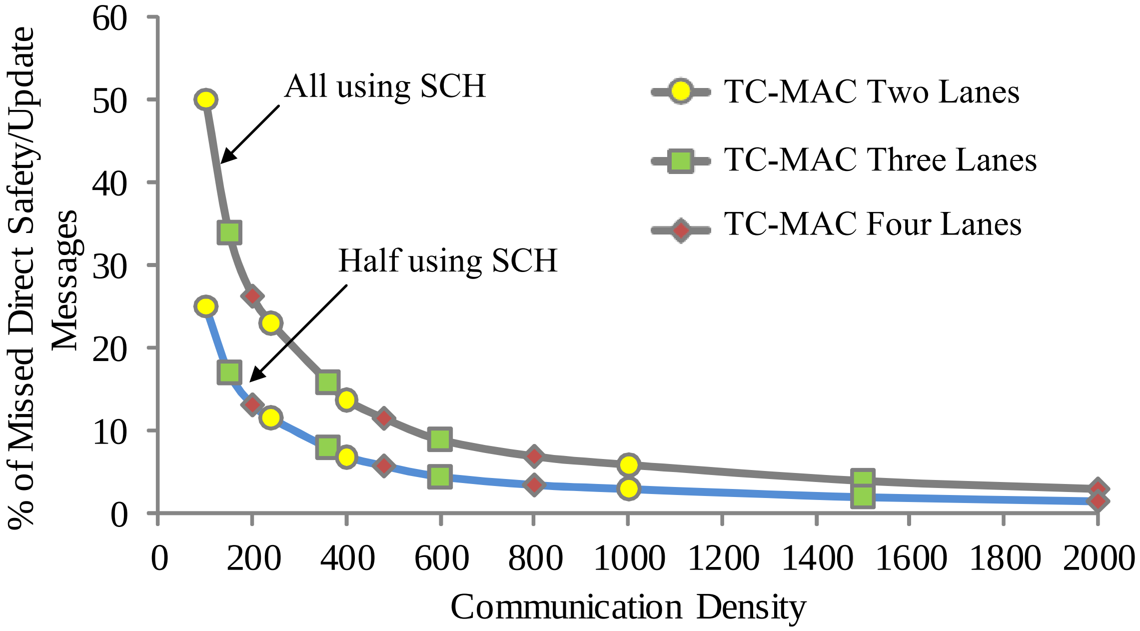


Fig. 42. Percentage of missed direct messages for TC-MAC in a single-hop cluster based on the communication density. It shows the performance when all vehicles and half vehicles in the cluster are engaged in communications during their own slot time on the *SCHs*

Throughput of Non-Safety Messages

For the throughput of non-safety messages, I tested TC-MAC and WAVE using the same scenarios as in the safety/update message communications. Every vehicle in the highway will try to send 1,200 Bytes every 100 *msec*. For TC-MAC, vehicles will use their own time slot on the SCHs, while vehicles in WAVE will try to send during the *SCHI*. Based on the network settings that we have, TC-MAC needs 1.6 *msec* to transmit a non-safety message. The *SCHI* value is set to 0.05 *sec*. Next, I will show the results for both TC-MAC and WAVE.

- TC-MAC

  From Figure 45, we can see the optimal calculated throughput of non-safety messages using TC-MAC. This happens when every pair of vehicles in the cluster that are engaged in communication on the *SCHs* has a different time slot number. The figure also shows the worst case calculated throughput of non-safety messages using TC-MAC; this happens only when every pair of vehicles in the cluster that are engaged in communication on the *SCHs* has the

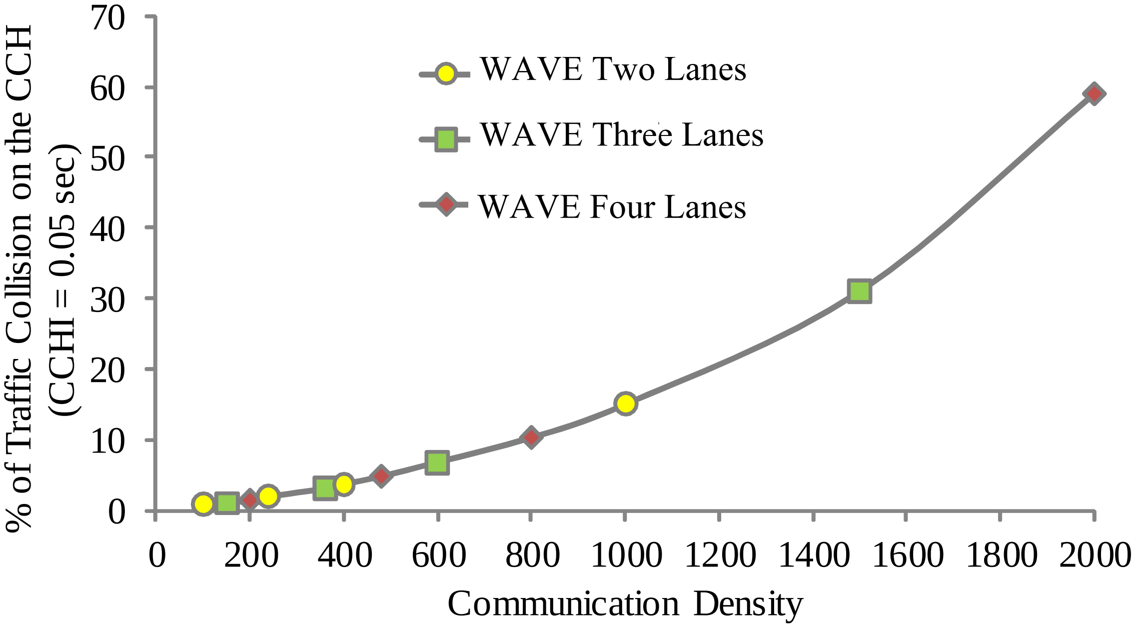


Fig. 43. Percentage of collisions during the *CCHI* for WAVE using a single-hop cluster

same time slot number. In the simulation, I used random pairs. Some of the pairs have the same time slot number in the *SCHs*, while the others do not. The results of the simulations shows that the performance of TC-MAC is in between the calculated values of the optimal and the worst throughput.

- WAVE

  Figure 46 shows the performance of WAVE compared to the worst case of TC-MAC. WAVE achieved a good throughput when the communication density of vehicles in the cluster was low. As the communication density goes higher, the throughput goes lower. The reason for that is due to the increase of the collision on the *SCHs*. Even when the communication density of the vehicles in WAVE was low, the best achieved throughput was lower that the worst calculated throughput in TC-MAC.

6.3.4 TWO-HOP CLUSTER EVALUATION

In the two-hop cluster, not all cluster members are in range of each other. So, not all vehicles can send and receive messages directly from other vehicles in the

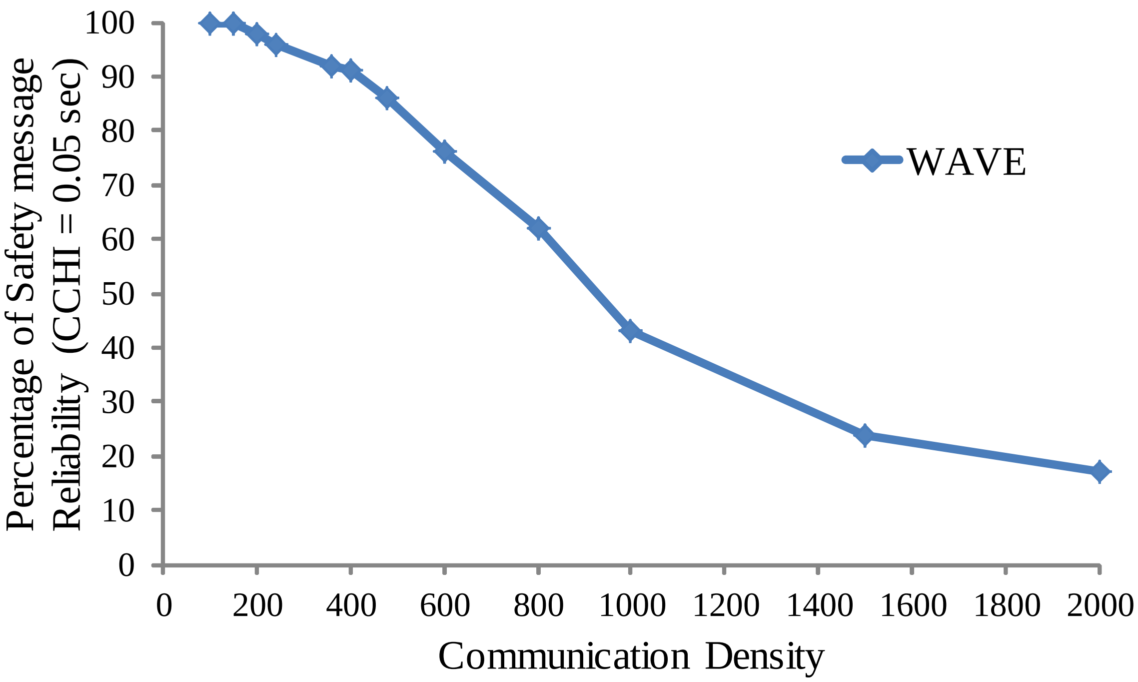


Fig. 44. Reliability of safety messages in WAVE using a single-hop cluster

same cluster. Now I will explain and show the performance of TC-MAC compared to WAVE in term of the reliability of safety/update. The throughput of non-safety messages in the two-hop cluster was not tested because that might involve a routing protocol, which is not the focus of this dissertation. Based on the transmission range used in the simulations, the two-hop cluster can cover an area of up to 600 m.

Table 10 shows the scenarios tested for TC-MAC and WAVE in testing the reliability of safety/update messages. I ran 5 runs for each scenario. I will show the results as the average of the runs for both protocols in following:

- TC-MAC

  Figure 47 shows the performance of TC-MAC using a two-hop cluster comparing to the communication density when only half of the vehicles in the cluster are using their slot on the *SCHs*. From the figure, it shows that TC-MAC has higher missed direct safety/update messages percentage in the two-hop cluster compared to the single-hop cluster. The reason for that is when calculating the percentage, all vehicles in the two-hop cluster are included even if they are out of range of other vehicles in the cluster. If the missed messages are safety messages, the *CH* will rebroadcast them to all vehicles in the cluster including

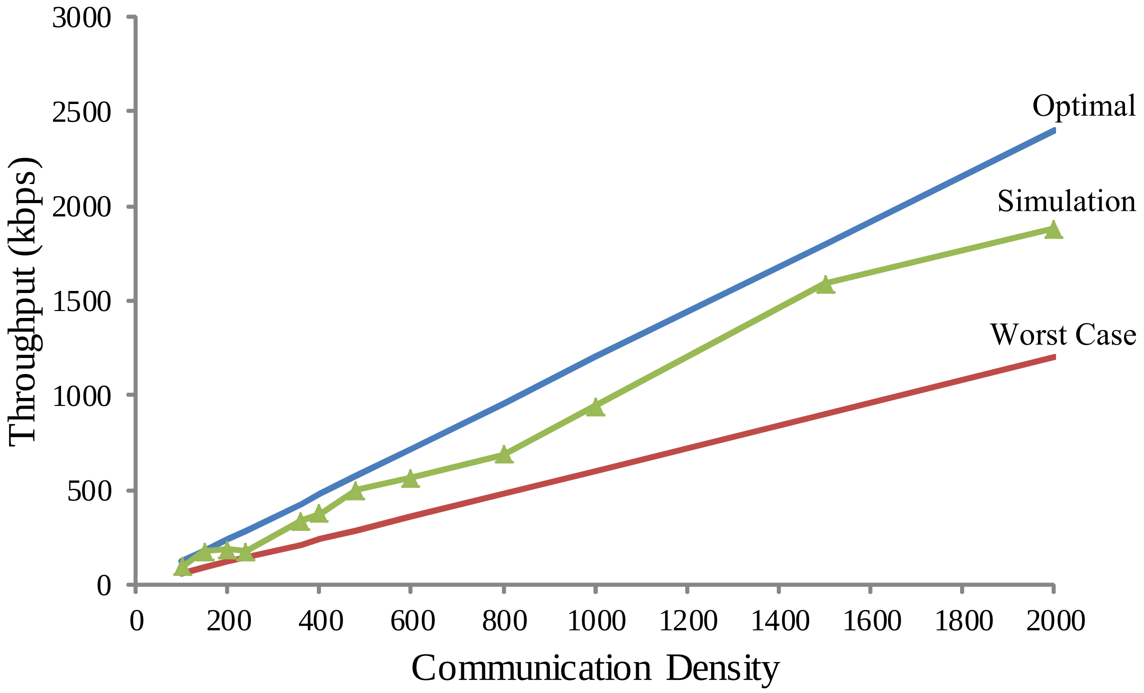


Fig. 45. Throughput of non-safety messages comparing to the communication density using TC-MAC in a single-hop cluster

the vehicles that missed them. On the other hand, if the missed messages are update messages, this should not be an issue because these messages are from vehicles that are more than one hop away.

- WAVE

  Figure 48 shows the performance of WAVE when the *CCHI* is 0.05 *sec* based on the Communication Density of vehicles in terms of traffic collision. From the figure, it is clear that as the Communication Density goes high, the percentage of traffic collision on the *CCH* goes high. The main reason for this issue is that every vehicle in the cluster is trying to compete to send its safety/update messages during the *CCHI*.

6.3.5 SUMMARY

I presented the TC-MAC protocol for intra-cluster communications in VANETs. I ran different simulations for TC-MAC along with WAVE to test the performance of TC-MAC compared to WAVE. TC-MAC showed that it can support higher reliability

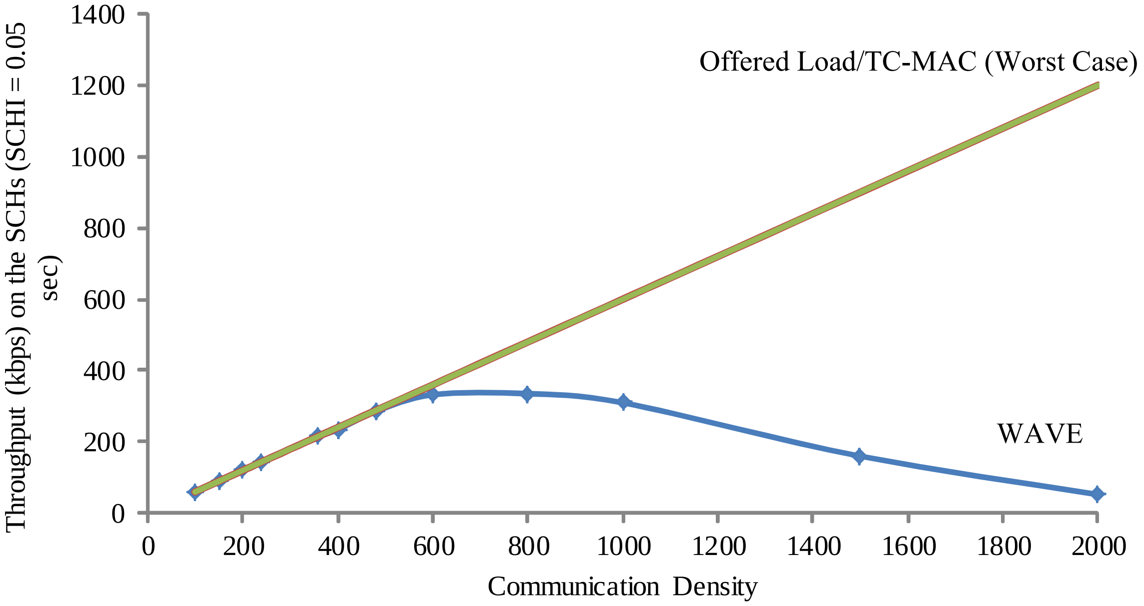


Fig. 46. Throughput of non-safety messages under different communication densities using WAVE in a single-hop cluster

of safety/update messages than WAVE standard, even in high density scenarios. Also, TC-MAC performed in collision free way by using TDMA. Not only were safety/update messages able to be delivered using TC-MAC, but non-safety messages also had a good throughput performance even when the traffic density was high. Every vehicle in TC-MAC has its own chance to perform non-safety communication in every 100 *msec*, without affecting their chances of sending/receiving safety/update messages. On the other hand, WAVE suffered from high traffic density. For WAVE, as the traffic density increases, the collision on the *CCH* increases.

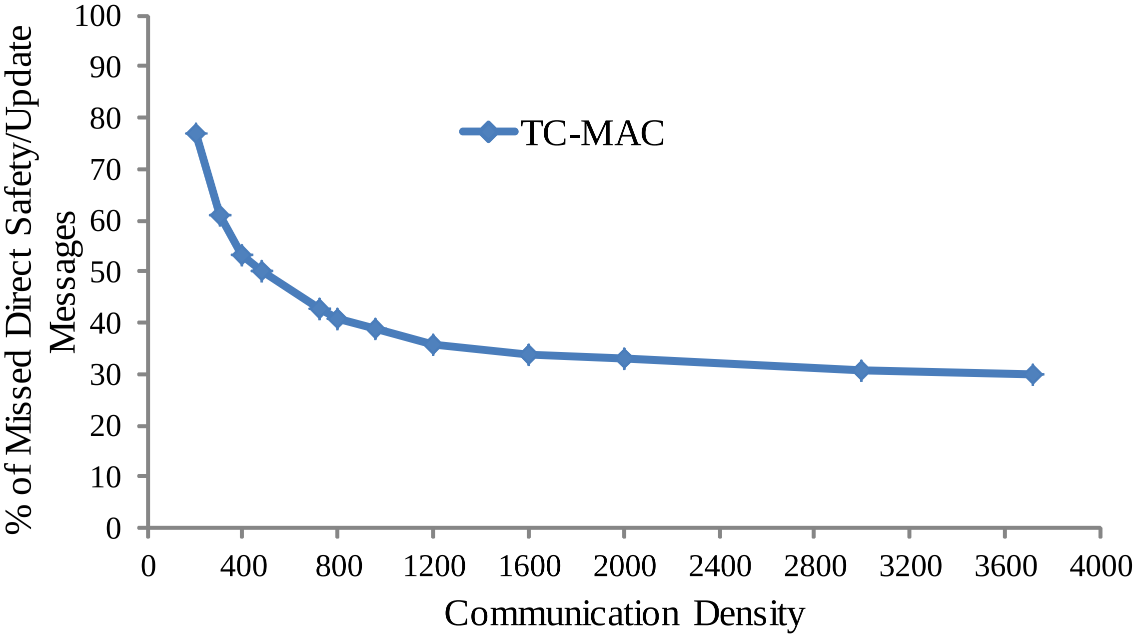


Fig. 47. Percentage of missed direct safety/update messages for TC-MAC in a two-hop cluster based on the communication density. Assuming that half of the vehicles in the cluster are engage in any sort of communications during their own slot time on the *SCHs*

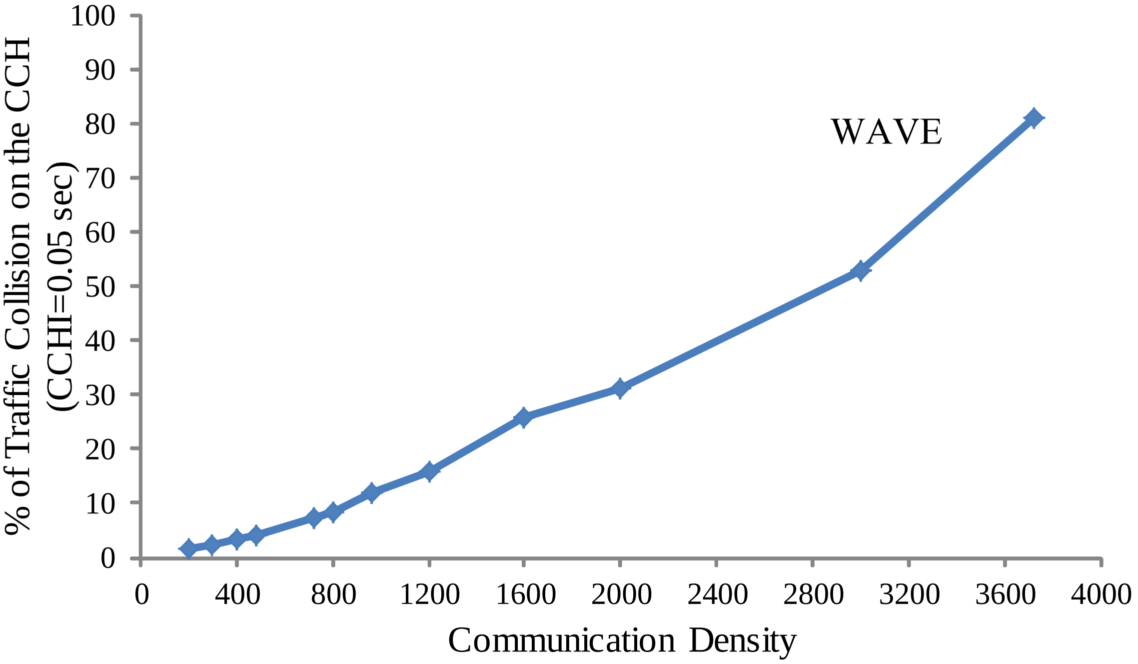


Fig. 48. Percentage of collisions safety/update messages during the *CCHI* for WAVE using a two-hop cluster

CHAPTER 7

APPLICATIONS USING TC-MAC

In this chapter I will describe a peer-to-peer (P2P) file sharing scheme for VANETs as an application that works under TC-MAC. This scheme aims to develop a P2P file sharing algorithm to improve the file downloading time between neighbouring vehicles.

The chapter is organized as follows. Section 1 gives a background of file sharing techniques in VANETs. Section 2 describes my P2P scheme in detail. Section 3 discusses the evaluation.

7.1 BACKGROUND OF FILE SHARING TECHNIQUES IN VANETS

The development of Peer-to-peer systems in VANET has been one of the hot topics in Vehicular Networks in the recent years. A number of the proposed systems for peer-to-peer rely on either on an existing (or imaginary) infrastructure or cellular system. Abuelela et al [59] introduced a zero-infrastructure peer-to-peer system for VANET (ZIPPER). ZIPPER is designed mainly to support multimedia streaming in VANETs such as movies and music. In CarTorrent [60], a work that extends the BitTorrent protocol to the vehicular networks scenarios, addressing issues such as intelligent peer and piece selection given the intermittent connectivity to preinstalled access point was proposed. Lee et al. [60] have implemented and deployed CarTorrent on a real VANET, which is the first implementation of a content sharing application on a real vehicular ad hoc test-bed. However, given the hundreds of highways miles at which there are hardly enough budget to maintain and install lights on the roads, installing gateways every 2-10 miles will be very expensive and not a practical solution. Liu et al [61] proposed Mobile Chord (MChord) which is an enhancement the P2P performance over Vehicular Ad hoc Network (VANET).

Various types of application that work on peer-to-peer systems can be implemented in VANET since peer-to-peer (P2P) is a powerful platform for a variety of multimedia streaming applications over the Internet such as video-on-demand, video conferencing and live broadcasting. Hossain et al. studied a case study of a peer-to-peer video conferencing system in VANET [62]. Hossain et al. distinguishes between active and passive participants and enhances the video quality of the active participant.

In PAVAN [63], a cellular network is used to broadcast a file description to all vehicles in a certain area. If a vehicle is interested in a file, a route should be discovered and maintained between it and the owner of the file. Scalability is an issue in PAVAN. AS the number of vehicles increases, the cellular network cannot handle all the requests and load of transmission.

7.2 P2P FILE SHARING IN VANETS USING TC-MAC

I propose a P2P file sharing scheme for VANETs on top of the TC-MAC protocol. The goal of the proposed work is to allow neighbouring vehicles to run non-safety applications such as large-scale file sharing and media streaming services in VANETs. I use the length of the TDMA frame as in TC-MAC, 100 *msec*. In this case, I can guarantee that every vehicle in the cluster sends one update/safety message every 100 msec to meet the safety message requirements.

To explain the P2P file sharing protocol using TC-MAC, suppose vehicle i wishes to share a large file with vehicle j; setting up a connection between them is done as follows:

1. By tuning in to vehicle j's own mini-slot, vehicle i determines whether or not vehicle j is available.
2. If so, vehicle i transmits a handshake packet on channel $j \ mod \ k$ during time slot $\lfloor \frac{j}{k} \rfloor$.
3. Since they are sharing a large file, vehicle i will ask a permission from the the CH to use other time slots on the SCHs.
4. CH will check for unused time slots on the SCHs and grant them to vehicles i and j. These granted time slots on the SCHs could be available because no vehicles assigned to their IDs, or because the vehicles assigned to them are un-active.
5. Now, vehicles i and j can start the transmission.

To ensure that vehicles i and j are still receiving update messages from other vehicles nearby during the transmission of the shared file, vehicles i and j will use

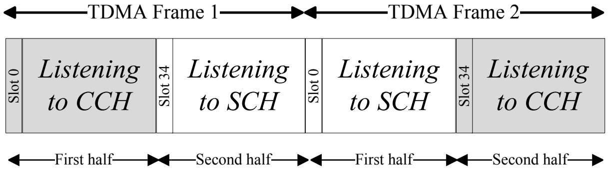


Fig. 49. The switching between two halves in one TDMA frame in P2P file sharing. The pair of vehicles that are involved in P2P file sharing will listen to the CCH in first half of the first TDMA frame and will use the time slots on the SCHs in the second half of the TDMA frame, and vice versa in the following TDMA frame

the granted time slots on the SCHs in the first half of the TDMA frame, and switch to the CCH during the second half of the TDMA frame. In the following TDMA frame, vehicles i and j will keep listening to the CCH, and then switch to the granted time slots on the SCHs in the second half of the TDMA frame. This process will continue until the file transmission is completed, or interrupted by the CH due to changes in the availability of the unused time slots (Figure 49).

Because vehicles in the process of transmitting a large file will switch between the two halves of the TDMA frame, they will only hear update/safety messages from other nearby vehicles every 200 *msec*. To solve this issue, we need to differentiate between the messages that are missed. If the missed messages are position update messages, the receiver can predict the movement of the sender during this time. On the other hand, if the missed messages are safety messages that are triggered by changes in vehicle behavior, the sender will collect feedback on its recent broadcast message from other vehicles and resend the safety message, if needed. This feedback is done using the Piggybacked Acknowledgement (PACK) protocol [41], which places the following information in each outgoing safety message:

- Sender's position
- The intended range of reception
- A randomly generated message ID
- IDs of most recently received messages (of which this sending node is within their intended ranges)
- The reception time ($time_{earliest}$) of the earliest message in the acknowledgement list

If vehicle i receives a message M_j from vehicle j, i is able to infer feedback on its recently transmitted message M_i if and only if two conditions are met: j is within the intended range of M_i, and the attached $time_{earliest}$ in M_j is earlier than the time M_i is sent.

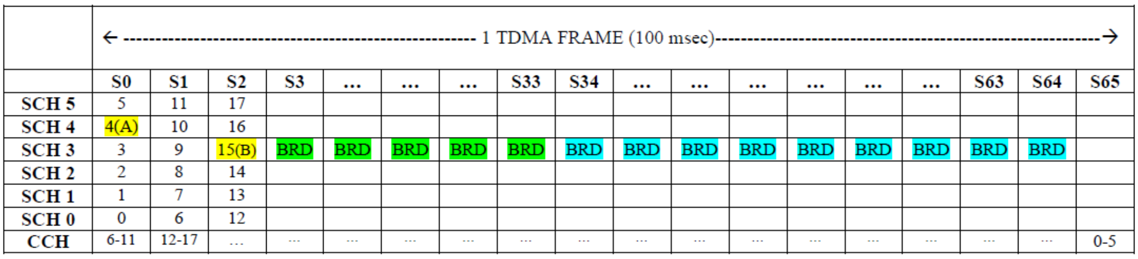
For an illustration, assume vehicle A with local ID=4 wants to share a 4 MB MP3 file with vehicle B with local ID=15. Vehicle A will make the handshake with vehicle B and will request time slots on the SCHs from the CH. Assuming that vehicle A and B will be granted the requested time slots, the transmission will take place as follow (Figure 50):

- Vehicle A will use (S1, SCH3) to send 1200 bytes to vehicle B.
- The CH will allow vehicles A and B to borrow slots from other cluster members. Assume all time slots are on SCH 3.
- In order for vehicles A and B to hear the surrounding vehicles, they will use the granted slots from one half of the TDMA frame and alternate with the other half in the following TDMA frame.
- Vehicles A will transfer data to vehicle B using slots from S2 to S33 on SCH 3.
- In the following frame, vehicle A and B will use slots from S34 to S64 on SCH3.
- During slot S65, vehicle A will switch to the CCH to broadcast update/safety messages to other cluster members.
- During slot S0, vehicle B will switch to the CCH to broadcast update/safety messages to other cluster members.

The total number of slots needed for the file to be transferred from vehicle A to vehicle B (Assuming the slot in SCH can send 1200 bytes of data) = the file size / slot size. If we have a file of size 4 MB, the vehicles need 3347 slots on the SCH to complete the transfer.

7.3 EVALUATION

To evaluate our P2P file sharing scheme, we assume we have a single-hop cluster, where vehicles can communicate with each other directly. The parameters for the network are listed in Table 11. We assume we have a full cluster; where all local IDs



	← -- 1 TDMA FRAME (100 msec)---→																	
	S0	**S1**	**S2**	**S3**	**…**	**…**	**…**	**S33**	**S34**	**…**	**…**	**…**	**…**	**…**	**…**	**S63**	**S64**	**S65**
SCH 5	5	11	17															
SCH 4	4(A)	10	16															
SCH 3	3	9	15(B)	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	BRD	
SCH 2	2	8	14															
SCH 1	1	7	13															
SCH 0	0	6	12															
CCH	6-11	12-17	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	0-5

Fig. 50. An example of two vehicles A and B sharing a file under TC-MAC. First half granted time slots are in green color and the second half are in blue color.

TABLE 11. Testing parameters

Parameters	Values
Cluster Length	300 m
TX Range	300 m
Safety Packet Size	200 bytes
Non-Safety Packet Size	1200 bytes
Data Rate	6 Mbps
Mini Slot Size	0.26 msec
SCH Slot Size	1.52 msec
Frame Size	100 msec
Number of Slots in the Frame	65
Shared File Size	4, 8, 12 MB

are assigned to vehicles in the cluster. We calculated the download time of a file for one pair of vehicles under different level of slots availabilities. Table 12 shows the percentage of the active vehicles in the cluster that I used in the simulation.

I evaluated the application through detailed simulation. We used the ns-3 network simulator [53], which is a follow-on to the popular ns-2 simulator. For VANETs, we used modules [54] that added well-known traffic mobility models, the Intelligent Driver Model (IDM) [55] and the MOBIL lane change model.

For our P2P file sharing scheme, the results in Figures 51, 52, and 53 show that even when all vehicles in the cluster are using their time slots to communicate, P2P file sharing still works but it takes longer time to download the file.

TABLE 12. Levels of active vehicles in the cluster

Slots Availability Level in SCH	Percentage of The Busy Slots on SCH	Number of Slots Borrowed
High	10	58
Med	50	32
Low	80	12
Very Low	100	0

For an illustration, we will show how we calculated the download time for a file in our P2P file sharing scheme. Assume we have 4 MB MP3 file to be shared between two vehicles in the cluster. Using the network setting in Table 11, the minimum time needed to transfer the file is 5.086 *sec*, or 3347 SCH slot times. Based on the size of the cluster, the activity of the cluster members, and the local IDs of the vehicles, the time needed to transfer a file may vary.

Let us assume we have two vehicles, A and B. If vehicle A and vehicle B are on different slot numbers on the SCH, and the cluster is filled with vehicles that are not using their slots on the SCHs, the time to download is calculated as follows:

Number of slots that vehicle A and vehicle B can listen to on the SCHs

= Number of slots in the TDMA frame - 2

= 66 - 2 = 64.

The reason we subtracted 2 is because vehicle A needs to switch to the CCH to send an safety/update message during its own mini-slot time, and this slot is different than vehicle B's mini-slot time. Since the P2P file sharing scheme uses one half of the available time slots in every TDMA frame, vehicles A and B will have only 32 slots on the SCH every TDMA frame. So, the total TDMA frames needed for the file to be transferred from vehicle A to vehicle B is:

Number of slots needed to transfer the file / number of usable slots in the TDMA frame

= 3347 / 32

= 105 frames. Since the frame is equal to 100 *msec*, the time needed to download a 4 MB file from vehicle A to vehicle B is 10.5 *sec*.

7.4 SUMMARY

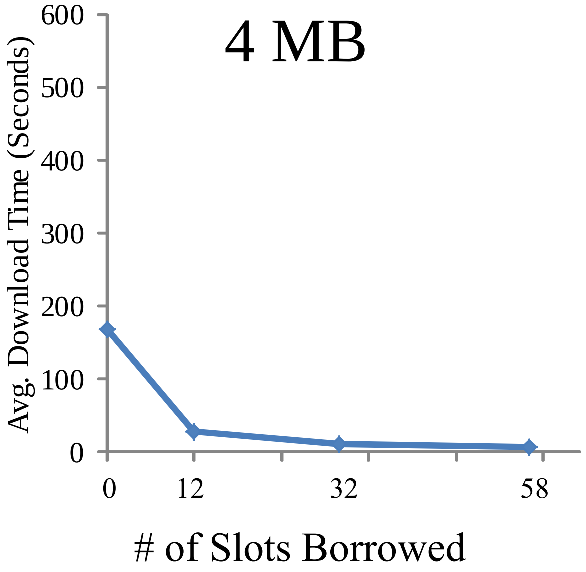


Fig. 51. Time to download 4 MB file using P2P file sharing scheme. There are a maximum of 58 slots available for borrowing.

In this chapter, we presented a P2P file sharing protocol using TC-MAC for VANETs. Unlike WAVE, when the number of vehicles that are involved in P2P file sharing is high, vehicles are still able to perform file sharing in each TDMA frame. We also explained the P2P file sharing scheme by using examples. The evaluation results shows that our scheme is able to file share between vehicles, as well as meeting the requirements of the safety messages.

In the future, we will further develop our scheme to have P2P file sharing between vehicles in different clusters. We are also interested in developing a better scheme to ensure the delivery of safety messages than what is proposed in the Piggyback Acknowledgement protocol.

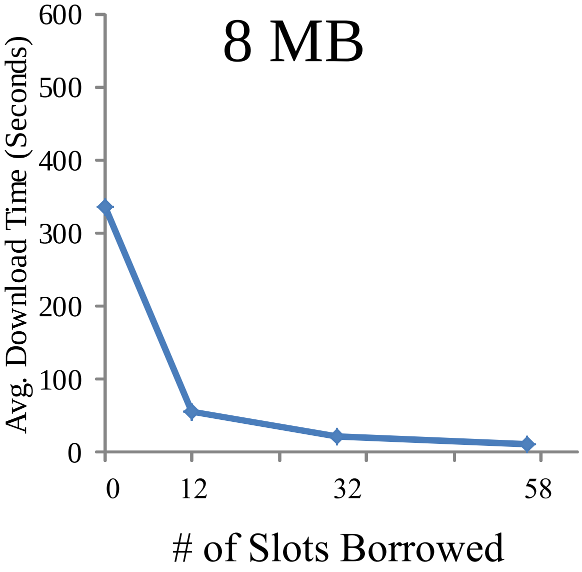


Fig. 52. Time to download 8 MB file using P2P file sharing scheme. There are a maximum of 58 slots available for borrowing.

Fig. 53. Time to download 12 MB file using P2P file sharing scheme. There are a maximum of 58 slots available for borrowing.

CHAPTER 8

CONCLUSION

In this chapter I summarize the motivation for this dissertation, the problem I have addressed, and the solutions I have proposed. The work in this dissertation also projects future research. The chapter is organized as follows: Section 1 addresses the summary and main motivation of this dissertation. Section 2 lists the contributions of this dissertation. Section 3 explains the proposed techniques for improving the communications in VANETs and their evaluations. Future extensions and developments of this work are shown in Section 4.

8.1 SUMMARY

Improving the safety and comfort of drivers and passengers by wirelessly exchanging information between vehicles and roadside units (RSUs) presents a major driving force for the design of Vehicular Ad-hoc Networks (VANETs). Several applications have been designed for VANETs, including safety and non-safety applications. Most, if not all, of the applications in VANETs require exchanging messages among vehicles and RSUs. There are three different types of messages in VANETs: periodic (update), event-driven (safety), and informational (non-safety) messages. Each type of these messages has its own usage, importance, priority, and generating rate.

Dedicated Short Range Communication (DSRC) is the 75 MHz wide spectrum band allocated by the U.S. Federal Communication Commission (FCC) for communications in VANETs. The spectrum band is divided into seven 10 MHz channels, one Control Channel (CCH), and six Service Channels (SCHs). The *CCH* is the default channel for the exchange of safety and update messages, while the *SCHs* are the default channels for non-safety.

The IEEE has completed the 1609 family of standards for Wireless Access in Vehicular Environments (WAVE) standard for vehicular communications. In WAVE, IEEE 1609.4 describes a concept of channel intervals in which time is divided into alternating Control Channel and Service Channel Intervals (*CCHI* and *SCHI*). The general concept calls for each interval to be 50 *msec* long. A pair of a *CCHI* and

SCHI forms a Sync Interval (SI) with the length of 100 *msec*, which is motivated by a desire of having a safety messages rate of 10 Hz. This desire is based on the allowable latency requirements of Life-Critical safety applications.

Unfortunately, channel switching in WAVE cannot support both safety and non-safety applications with high reliability at high traffic densities. Either safety applications or non-safety applications must be compromised. Since safety applications have higher priority than non-safety applications, and to maintain the 100 *msec* requirement of safety messages, non-safety applications may be shut down.

This dissertation has proposed vehicular communication protocols to overcome existing challenges and support safety and non-safety applications. I have designed a Cluster-based Medium Access Control (MAC) protocol for communications in VANETs that uses the Time Division Multiplexing Access (TDMA) technique.

8.2 CONTRIBUTIONS

In this dissertation, I have made the following contributions:

- *A multi-channel cluster-based TDMA MAC protocol to coordinate intra-cluster communications (TC-MAC).* The proposed protocol can be used for clustering management and communications. This protocol integrates the centralization approach of clusters and a new scheme for slot reservation, using cluster members' local IDs. In this technique, all vehicles are able to tune to the *CCH* or one of the *SCHs* if needed during the time cycle. In other words, the time cycle is not divided into two different intervals, *CCHI* and *SCHI*, as with WAVE.

- *A CH election and cluster formation algorithm based on the traffic flow and design a cluster maintenance algorithm that benefits from our cluster formation algorithm.* Rather than considering some of VANET characteristics in the election of the *CH*, my algorithm puts into account the traffic flow on the road. The design and implementation of this *CH* election and cluster formation algorithm shows fewer *CH* changes, which reduces the overhead of re-clustering and delivers an efficient hierarchical network topology. During the cluster formation process, the cluster members will be assigned local IDs by the *CH*. Vehicles in VANETs are allowed to move freely, therefore, I propose a new cluster maintenance algorithm that handles topology changes caused by mobility changes. The proposed algorithm takes advantage of the local IDs that are assigned in our cluster formation algorithm.

- *A multi-channel cluster-based CDMA/TDMA hybrid MAC protocol to coordinate inter-cluster communications.* I propose a MAC protocol that enables vehicles to communicate with vehicles in different clusters and with *RSUs*. In addition, the hidden and exposed terminal problems are addressed by the proposed protocol.

8.3 EVALUATION

The evaluation was performed in the ns-3 network simulator. For VANETs, I used modules that added well-known traffic mobility models, the Intelligent Driver Model (IDM) and the MOBIL lane change model. The goal was to create a vehicular network on highways with different number of lanes and different number of vehicles.

8.3.1 CLUSTERHEAD ELECTION

I evaluated the algorithm for clusterhead election based on the traffic flow of vehicles in the highway. With the availability of lane detection, lane direction and map matching, this algorithm was able to select the most stable clusterhead. I tested this algorithm using a highway with a two exit scenario and followed the elected clusterhead passing the two exits. This algorithm showed longer clusterhead lifetime than the Lowest-ID, Highest-Degree and the Utility Function algorithms.

8.3.2 TC-MAC

I evaluated the TC-MAC protocol for intra-cluster communications in VANETs. I ran different simulations for TC-MAC along with WAVE to test the performance of TC-MAC compared to WAVE. TC-MAC showed that it can support higher reliability of safety/update messages than WAVE standard, even in high density scenarios. Also, TC-MAC performed in collision free way by using TDMA. Not only were safety/update messages able to be delivered using TC-MAC, but non-safety messages also had a good throughput performance even when the traffic density was high. Every vehicle in TC-MAC has its own chance to perform non-safety communication in every 100 *msec*, without affecting their chances of sending/receiving safety/update messages. On the other hand, WAVE suffered from high traffic density. For WAVE, as the traffic density increases, the collision on the *CCH* increases.

8.4 FUTURE WORK

From this dissertation, there are some clear directions for the future work. These directions can be classified as further analysis of TC-MAC, designing a numbering scheme for cluster members, enhancing the utilization on the *SCHs*, and developing speed-based clustering.

As in all evaluations, there is more analysis that can be performed. I would like to further investigate the effect of guard intervals (GIs) on the number of cluster members and message size.

For the numbering scheme for cluster members, the *CH* in TC-MAC assigns the local IDs to the cluster members starting from ID 2 to $N_{max} - 1$. These IDs are assigned in order. So, if there are eleven vehicles in the cluster, the vehicles will have IDs from 1 to 11. In this way, vehicles with IDs 6 to 11 will have their mini-slots during *S0* of the TC-MAC frame. For vehicles with IDs 1 to 5, their time slots on the *SCHs* will be during *S0* of the TC-MAC frame as well. So, if any vehicle with IDs 1 to 5 is communicating on its own slot on the *SCHs*, this vehicle will miss all the update messages from the other cluster members on the *CCH*. The new numbering scheme should make the percentage of missed update messages less than the current scheme. This can be achieved by giving the cluster members local IDs in a way that makes the number of vehicles that are sharing the time slot on the *SCHs* as minimum as possible. On other words, the *CH* should start assigning IDs from the slots that have no vehicles using them on the *SCHs*. If all slots on the *SCHs* has at least one vehicle using it, the *CH* will start assigning IDs from the slots that have the least number of vehicles. This process continues until the cluster reaches its maximum capacity. The peer-to-peer file sharing protocol, discussed in Chapter 7, shows some enhancements on utilizing the *SCHs*. What I would like to do is to improve the utilization on the *SCHs* even more to make it as close as possible to the optimal values. Also, I would like to study the impact of the enhancement on the safety/update messages reliability and the overhead. The clustering formation algorithm I used in this dissertation is based on the highways in the U.S., where all the lanes on the highway have the same speed limit. However, in other countries, the highway may have different speeds based on the lane. So, using the same clustering formation for such highways will make the process of new vehicles joining and leaving the cluster more frequent, which may lead to an increase overhead. I would like to investigate the impact of using a new clustering formation algorithm on the performance of TC-MAC.

REFERENCES

[1] S. Yang, H. Refai, and X. Ma, “CSMA based inter-vehicle communication using distributed and polling coordination,” in *Proc. of the 8th IEEE Conference on Intelligent Transportation Systems (ITSC ’05)*, Sept. 2005, pp. 167 – 171.

[2] “IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operation,” *IEEE Std 1609.4-2006*, 2006.

[3] “Draft Amendment to Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications: Wireless Access in Vehicular Environments,” *IEEE WG, IEEE 802.11p/D2.01*, 2007.

[4] F. Yu and S. Biswas, “Self-Configuring TDMA Protocols for Enhancing Vehicle Safety With DSRC Based Vehicle-to-Vehicle Communications,” *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1526–1537, 2007.

[5] H. A. Omar, W. Zhuang, and L. Li, “VeMAC: A novel multichannel MAC protocol for vehicular ad hoc networks,” in *Proc. of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr. 2011, pp. 413 –418.

[6] World Health Organization. [Online]. Available: http://www.who.int/features/factfiles/roadsafety/en/index.html

[7] D. Schrank, T. Lomax, and S. Turner, “2010 Annual Urban Mobility Report,” Texas Transportation Institute, The Texas A&M University System, Tech. Rep., 2010.

[8] United States Department of Transportation. Intelligent Transportation Systems. [Online]. Available: www.its.dot.gov/index.htm

[9] A. E2213-03, “Standard specification for telecommunications and information exchange between roadside and vehicle systems – 5 GHz band dedicated short range communications (DSRC) medium access control (MAC) and physical layer (PHY) specifications,” *ASTM International*, July 2003.

[10] “IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Architecture,” *IEEE Std 1609.0-2007*, 2007.

[11] Q. Chen, D. Jiang, and L. Delgrossi, "IEEE 1609.4 DSRC multi-channel operations and its implications on vehicle safety communications," in *Proc. of the 2009 IEEE Vehicular Networking Conference (VNC).* IEEE, Oct. 2009, pp. 1–8.

[12] Z. Wang and M. Hassan, "How much of DSRC is available for non-safety use?" in *Proc. of the fifth ACM international workshop on VehiculAr Inter-NETworking*, ser. (VANET), 2008, pp. 23–29.

[13] J. J. Blum, A. Eskandarian, and L. J. Hoffman, "Challenges of Intervehicle Ad Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 347 – 351, Dec. 2004.

[14] M. S. Almalag, "Safety-related vehicular applications," in *Vehicular Networks From Theory to Practice*, S. Olariu and M. C. Weigle, Eds. Chapman & Hall/CRC, 2009.

[15] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," *Wireless Network*, vol. 8, no. 2/3, pp. 153–167, Mar. 2002.

[16] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety," *Communications Magazine, IEEE*, vol. 44, no. 1, pp. 74–82, Jan. 2006.

[17] R. Chen, W.-L. Jin, and A. Regan, "Broadcasting Safety Information in Vehicular Networks: Issues and Approaches," *IEEE Network*, vol. 24, no. 1, pp. 20–25, 2010.

[18] Verizon wireless. [Online]. Available: http://www.verizonwireless.com/

[19] "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Resource Manager," *IEEE Std 1609.1-2006*, 2006.

[20] "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Security Services for Applications and Management Messages," *IEEE Std 1609.2-2006*, 2006.

[21] "IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Networking Services," *IEEE Std 1609.3-2007*, 2007.

[22] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments," in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, may 2008, pp. 2036 –2040.

[23] M. Garg and R. K. Shyamasundar, "A Distributed Clustering Framework in Mobile Ad Hoc Networks," in *International Conference on Wireless Networks*, 2004, pp. 32–38.

[24] G. Wolny, "Modified DMAC Clustering Algorithm for VANETs," in *Systems and Networks Communications, 2008. ICSNC '08. 3rd International Conference on*, oct. 2008, pp. 268 –273.

[25] X. Hong, K. Xu, and M. Gerla, "Scalable Routing Protocols for Mobile Ad Hoc Networks," *Network, IEEE*, vol. 16, no. 4, pp. 11 –21, jul/aug 2002.

[26] Y. Gunter, B. Wiegel, and H. Grossmann, "Cluster-based Medium Access Scheme for VANETs," in *Proc. IEEE Intelligent Transportation Systems Conference*, Oct. 2007, pp. 343 –348.

[27] M. Jiang, J. Li, and Y. C. Tay, "Cluster based routing protocol(CBRP)," pp. 1 –27, Aug. 1999.

[28] P. Fan, J. G. Haran, J. Dillenburg, and P. C. Nelson, "Cluster-Based Framework in Vehicular Ad-Hoc Networks," 2005, pp. 32 – 42.

[29] Z. Rawshdeh and S. Mahmud, "Toward Strongley Connected Clustering Structure in Vehicular Ad Hoc Networks," in *Proc. of the 70th IEEE Vehicular Technology Conference (VTC 2009-Fall)*, Sept. 2009, pp. 1 –5.

[30] G. Venkataraman, S. Emmanuel, and S. Thambipillai, "Size-restricted cluster formation and cluster maintenance technique for mobile ad hoc networks," *Int. J. Netw. Manag.*, vol. 17, no. 2, pp. 171–194, Mar. 2007.

[31] L. Wang and S. Olariu, "Cluster Maintenance in Mobile Ad-hoc Networks," *Cluster Computing*, vol. 8, no. 2-3, pp. 111–118, 2005.

[32] L. Bononi and M. Di Felice, "A Cross Layered MAC and Clustering Scheme for Efficient Broadcast in VANETs," in *Proc. IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, Oct. 2007, pp. 1 – 8.

[33] Z. Rawashdeh and S. Mahmud, "Media Access Technique for Cluster-Based Vehicular Ad Hoc Networks," in *Proc. IEEE 68th Vehicular Technology Conference VTC*, Sept. 2008, pp. 1 – 5.

[34] F. Farnoud and S. Valaee, "Repetition-based Broadcast in Vehicular Ad Hoc Networks in Rician Channel with Capture," in *Proc. the 27th Conference on Computer Communications (INFOCOM '08)*, Apr. 2008, pp. 1 – 6.

[35] B. Hassanabadi, L. Zhang, and S. Valaee, "Index Coded Repetition-Based MAC in Vehicular Ad-Hoc Networks," in *Proc. of the 6th IEEE Conference on Consumer Communications and Networking Conference*, Jan. 2009, pp. 1180–1185.

[36] O. Kayis and T. Acarman, "Clustering Formation for Inter-Vehicle Communication," in *Proc. IEEE Intelligent Transportation Systems Conference, 2007. (ITSC '07)*, Oct. 2007, pp. 636 – 641.

[37] H. Su and X. Zhang, "Clustering-Based Multichannel MAC Protocols for QoS Provisionings Over Vehicular Ad Hoc Networks," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3309 – 3323, Nov. 2007.

[38] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*, 5th ed. USA: Addison-Wesley Publishing Company, 2009.

[39] P. Djukic and P. Mohapatra, "Soft-TDMAC: A software TDMA-based MAC over commodity 802.11 hardware," in *Proc. of IEEE INFOCOM*, Apr. 2009, pp. 1836 – 1844.

[40] C. Suthaputchakun and A. Ganz, "Priority Based Inter-Vehicle Communication in Vehicular Ad-Hoc Networks using IEEE 802.11e," in *Proc. IEEE 65th Vehicular Technology Conference, (VTC '07)*, Apr. 2007, pp. 2595 – 2599.

[41] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich, "Design of 5.9 GHz DSRC-Based Vehicular Safety Communication," *Wireless Communications, IEEE*, vol. 13, no. 5, pp. 36 – 43, Oct. 2006.

[42] V. Taliwal and D. Jiang, "Mathematical Analysis of IEEE 802.11 Broadcast Performance in a Probabilistic Channel," DaimlerChrysler Technical Paper, Tech. Rep., 2005.

[43] Y. Bi, K.-H. Liu, L. Cai, X. Shen, and H. Zhao, “A Multi-Channel Token Ring Protocol for QoS Provisioning in Inter-Vehicle Communications,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 11, pp. 5621 – 5631, Nov. 2009.

[44] F. Herzel, G. Fischer, and H. Gustat, “An integrated CMOS RF synthesizer for 802.11 a wireless LAN,” *IEEE Journal of Solid-State Circuits*, vol. 38, no. 10, pp. 1767–1770, 2003.

[45] *NAVTEQs NAVSTREETS Street Data Reference Manual*, 3rd ed., Oct. 2009.

[46] W. Holfelde, “Vehicle safety communications in the US,” in *ITS World Congress*, London, U.K., 2007.

[47] A. J. Dean and S. N. Brennan, “Terrain-based road vehicle localization on multi-lane highways,” in *Proc. of the 2009 conference on American Control Conference*, ser. ACC’09. IEEE Press, 2009, pp. 707–712.

[48] R. Aufrere, R. Chapuis, and F. Chausse, “A model-driven approach for real-time road recognition,” *Mach. Vis. Appl.*, vol. 13, no. 2, pp. 95–107, 2001.

[49] M. Bertozzi and A. Broggi, “GOLD: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection,” *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 7, pp. 62–81, 1998.

[50] T. Oskiper, Z. Zhu, S. Samarasekera, and R. Kumar, “Visual odometry system using multiple stereo cameras and inertial measurement unit,” in *Proc. IEEE Conference on CVPR07*, 2007.

[51] M. Jabbour, P. Bonnifait, and V. Cherfaoui, “Enhanced Local Maps in a GIS for a Precise Localisation in Urban Areas,” in *9th IEEE Conference on Intelligent Transportation Systems (ITSC 2006)*, 2006.

[52] C. D. McGillem and T. S. Rappaport, “A beacon navigation method for autonomous vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 38, no. 3, pp. 132–139, 1989.

[53] T. R. Henderson, S. Roy, S. Floyd, and G. F. Riley, “ns-3 project goals,” in *Proc. of from the workshop on ns-2: the IP network simulator*, 2006.

[54] H. Arbabi and M. C. Weigle, “Highway mobility and vehicular ad-hoc networks in ns-3,” in *Proc. of the 2010 Winter Simulation Conference (WSC)*, Dec. 2010, pp. 2991 –3003.

[55] M. Treiber, A. Hennecke, and D. Helbing, “Congested traffic states in empirical observations and microscopic simulations,” *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, Aug. 2000.

[56] A. Kesting, M. Treiber, and D. Helbing, “General Lane-Changing Model MOBIL for Car-Following Models,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1999, pp. 86–94, Dec. 2007.

[57] A. Weimerskirch, J. J. Haas, Y.-C. Hu, and K. P. Laberteaux, “Data Security in Vehicular Communication Network,” in *VANET: vehicular applications and inter-networking technologies*, H. Hartenstein and K. P. Laberteaux, Eds. Wiley Online Library, 2010.

[58] D. Jiang, Q. Chen, and L. Delgrossi, “Communication Density: A Channel Load Metric for Vehicular Communications Research,” in *Proc. IEEE 4th International Conference on Mobile Adhoc and Sensor Systems, (MASS ’07).* IEEE, 2007, pp. 1–8.

[59] M. Abuelela and S. Olariu, “Zipper: A Zero-Infrastructure Peer-To-Peer System For VANETs,” in *Proc. of the ACM Workshop on Wireless Multimedia Networking and Performance Modeling (WMuNeP)*, Sep. 2007, pp. 2–8.

[60] K. Lee, S.-H. Lee, R. Cheung, U. Lee, and M. Gerla, “First Experience with CarTorrent in a Real Vehicular Ad Hoc Network Testbed,” in *Proc. IEEE Mobile Networking for Vehicular Environments Conference*, May 2007, pp. 109 – 114.

[61] C.-L. Liu, C.-Y. Wang, and H.-Y. Wei, “Cross-layer Mobile Chord P2P protocol design for VANET,” *IJAHUC*, vol. 6, no. 3, pp. 150–163, 2010.

[62] T. Hossain, Y. Cui, and Y. Xue, “VANETs: Case Study of a Peer-to-peer Video Conferencing System,” in *Proc. of the 6th IEEE Conference on Consumer Communications and Networking Conference*, ser. CCNC’09, 2009, pp. 54–55.

[63] S. Ghandeharizade, S. Kapadia, and B. Krishnamachari, “PAVAN: a policy framework for content availabilty in vehicular ad-hoc networks,” in *Proc. of the*

1st ACM international workshop on Vehicular ad hoc networks, ser. VANET '04. ACM, 2004, pp. 57–65.

VITA

Mohammad Salem Almalag
Department of Computer Science
Old Dominion University
Norfolk, VA 23529

Mohammad Almalag was a top-ranked student in the field of computer science at the Computer and Information Systems College. He received his Bachelors Degree in Computer Science from King Saud University in 2000. He was also recognized as the top-ranked student of the class of 2000. In Spring 2001, he joined the Department of Computer Science and Information System at Imam University as a teacher. In Fall 2002, Mohammad joined the masters program in the Department of Computer Science at Ball State University. He received his Masters Degree in Computer Science in Fall 2004. Mohammad joined the PhD program in the Department of Computer Science at Old Dominion University on Fall 2005. On 2008, he started his research under the supervision of Dr. Michele C. Weigle. Mohammad had been active in the areas of wireless networks, sensor networks, vehicular networks, MAC layer protocols, and networks simulation. With interest in MAC layer protocols and vehicular networks, Mohammad's PhD focus, including his dissertation, was on developing a new MAC protocol for vehicular ad-hoc networks. Specifically, his focus was on improving the non-safety application share of the bandwidth.

Typeset using LaTeX.